THE INTERPLAY OF BIOMECHANICS, TISSUE POLARITY AND COLLECTIVE MIGRATION AS IT CONTRIBUTES TO EARLY HEART ORGANOGENESIS

by

Timothy Ryan Jackson

B.S. Biological Sciences, University of Pittsburgh, 2009

B.Phil. Religious Studies, University of Pittsburgh, 2009

Submitted to the Graduate Faculty of

Swanson School of Engineering in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

University of Pittsburgh

2017

UNIVERSITY OF PITTSBURGH

SWANSON SCHOOL OF ENGINEERING

This dissertation was presented

by

Timothy Ryan Jackson

It was defended on

March 15, 2017

and approved by

Sanjeev Shroff, PhD, Distinguished Professor and Gerald McGinnis Chair
Departments of Bioengineering and Medicine

James H-C. Wang, PhD, Professor
Departments of Orthopedic Surgery and Bioengineering

Lei Yang, PhD, Associate Professor,
Departments of Developmental Biology and Bioengineering

Dissertation Director: Lance A. Davidson, PhD, Associate Professor,
Departments of Bioengineering, Developmental Biology, Computational and Systems Biology

THE INTERPLAY OF BIOMECHANICS, TISSUE POLARITY AND COLLECTIVE MIGRATION AS IT CONTRIBUTES TO EARLY HEART ORGANOGENESIS

Timothy Ryan Jackson, PhD

University of Pittsburgh, 2017

In early heart development, bilateral fields of heart progenitor cells (HPCs) undergo a large-scale movement from the anterior lateral plate mesoderm to merge on the ventral midline, undergoing a mesenchymal-to-epithelial transition (MET) halfway through this process. While the heart is the first functioning organs in the developing embryo, a comprehensive model for early heart development that integrates both physical mechanisms and molecular signaling pathways remains elusive. Here, we utilize Xenopus embryos to investigate the role of mechanical cues in driving MET in HPCs and show how dysregulation of these cues can cause congenital heart defects (CHDs).

Small molecule inhibitor treatments targeting actomyosin contractility reveal a temporally specific requirement of bulk tissue compliance to regulate heart development and MET. Through tracking of tissue level deformations in the heart forming region (HFR) as well as movement trajectories and traction generation of individual HPCs, we find the onset of MET correlates with a peak in mechanical stress within the HFR and changes in HPC migratory behaviors. Targeting mutant constructs to modulate contractility and compliance in the underlying endoderm, we find MET in HPCs can be accelerated in response to microenvironmental stiffening and can be inhibited by softening. To test whether MET in HPCs

is responsive to purely physical mechanical cues, we mimicked a high stress state by injecting an inert oil droplet to generate high strain in the HFR, demonstrating that exogenously applied stress is sufficient to drive MET. MET-induced defects in anatomy result in defined functional lesions in the larval heart and furthermore, when we recreate a clinically-relevant CHD phenotype through overexpression of a Noonan Syndrome-associated mutant protein, we find it leads to abnormal MET in HPCs due to a decoupling of force transmission and mechanosensory pathways.

From this integrated analysis of HPC polarity and mechanics, we propose that normal heart development requires HPCs to undergo a critical behavioral and phenotypic transition on their way to the ventral midline and that this transition is driven in response to the changing mechanical properties of their endoderm substrate. We conclude that the etiology underlying many CHDs may involve errors in mechanical signaling and MET.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

PREFACE

I would like to thank my PhD committee members for their valuable feedback and guidance. In particular, thank you to my advisor Dr. Lance Davidson, whose mentorship, expertise, creativity and encouragement empowered me to navigate through graduate school and the greater field of scientific research. Thank you to all of my fellow lab mates, for the troubleshooting tips, technical help, ideas, perspectives and morale boosting. And of course, thank you to my family and friends for the love and support you have provided.

The following acronyms are used frequently throughout the manuscript:

ALPM = Anterior lateral plate mesoderm

HFR = Heart forming region

HPCs = Heart progenitor cells

MET = Mesenchymal-to-epithelial transition

TFM = Traction-force microscopy

NS = Noonan syndrome

1.0 INTRODUCTION

1.1 *XENOPUS* HEART DEVELOPMENT

The process of heart development is highly conserved among vertebrates, particularly at early stages (Warkman and Krieg 2007). In amniotes, heart progenitor cells (HPCs) derive from a posterior portion of the epiblast just lateral to the primitive streak (Lyons 1996, Evans, Yelon et al. 2010). These cells are then among the first involute through the streak and move laterally into the lateral plate mesoderm in chick or the cardiac crescent in mouse (Evans, Yelon et al. 2010). In *Xenopus* heart development, HPCs can be fate mapped as early as the 32-cell stage embryo, where they derive from one of the dorsal vegetal blastomeres (Fig. 1A; (Dale and Slack 1987)). Prior to gastrulation, heart progenitor cells (HPCs) flank either side of the Spemann organizer at the dorsal blastopore lip (Fig. 1B; (Fishman and Chien 1997)). Then during gastrulation, heart progenitor cells (HPCs) are again among the first to involute through the blastopore, forming bilateral fields within the anterior lateral plate mesoderm (Fig. 1C). It is thought that HPCs are first induced by the Spemann organizer and following gastrulation they have already committed to a cardiac fate (Fishman and Chien 1997), although the presence of endoderm is also required to continue that induction process (Nascone and Mercola 1995). Wnt antagonist expression at the beginning of gastrulation is required to induce cardiogenesis, particularly Dkk1 and Crescent (Schneider and Mercola 2001), and BMP and FGF signaling families have also been shown to

regulate a number of cardiac transcription factors (Harvey 2002). The earliest markers for HPCs are cardiac transcription factors such as Nkx2.5, and GATA6 and Tbx1, which are expressed shortly after gastrulation at Stage 13 (Gessert and Kuhl 2009).

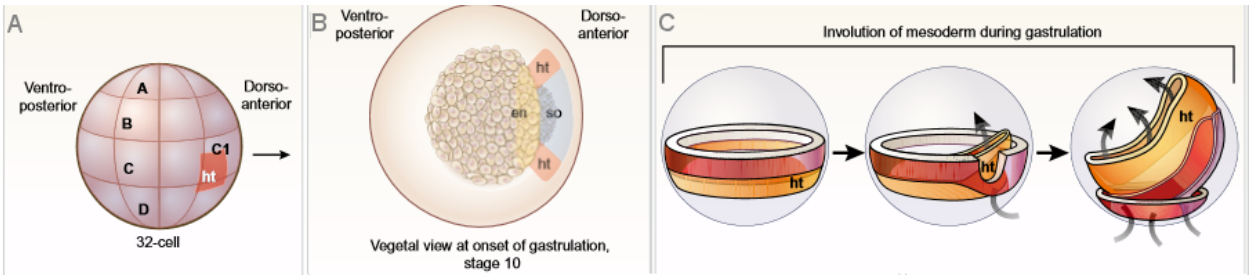

Figure 1. HPC location during early *Xenopus* development.

Schematic of HPC location (A) in the 32-cell stage embryo, (B) at the dorsal blastopore lip at the onset of gastrulation, and (C) during mesoderm involution in early gastrulatio. ht = HPCs, so = Spemann's organizer, en = endoderm. From Mercola et al., 2010.

In the early neurula, bilateral HPCs residing in the anterior lateral plate mesoderm (ALPM) begin moving ventrally (Fig. 2A-B). Although HPCs derive from one common cell population, they are thought to subdivide into distinct "fields" (Gessert and Kuhl 2009). These subpopulations are often called the first heart field, which fuses at the midline to form the myocardium and endocardium, and the secondary heart field, which assembles onto the arterial and venous poles of the developing heart tube and is often identified by the expression of the Islet family of transcription factors (Fig. 2D-E; (Brade, Gessert et al. 2007, Witzel, Cheedipudi et al. 2017)). Regardless, HPCs move to the ventral midline in close association with the endoderm. Classically, it was thought the HPCs migrate over the endoderm, which in turn serves to provide growth factors and act as a substrate (Trinh and Stainier 2004). However, recent studies have shown the endoderm plays a direct role in HPC movements by contracting to bring HPCs to the

ventral heart forming region (HFR; (Varner and Taber 2012, Shi, Varner et al. 2015, Ye, Xie et al. 2015)). During their movements to the HFR, HPCs undergo a mesenchymal-to-epithelial transition (MET), demarcated by expression of epithelial markers like aPKC, polarized ion channels and formation of tight junctions. (Linask 1992, Yelon and Stainier 2005). HPCs move as a bilayer, which eventually delaminates with the deeper layer contributing to the myocardium and endocardium and the more superficial layer contributing to the pericardium (Fig. 2D; (Mohun, Leong et al. 2000)). Upon reaching the HFR at Stage 27-28, the two populations of HPCs merge and fold to form a heart tube and proceed to later stages of heart development (Fig. 2C). At this point, cardiomyocytes begin to express smooth muscle markers, like tropomyosin (Kolker, Tajchman et al. 2000). This movement from the ALPM to the ventral heart forming region (HFR) will be the focus of this dissertation. The process, which involves an MET halfway through, lays the foundation for all later stages of heart development. While correct migration is not required for terminal differentiation of HPCs (Li et al., 2004), it is likely necessary for HPC organization into coherent epithelia. Incomplete migration can cause cardia bifida, chamber malformation, hypertrophy, and other defects (Bartlett et al., 2007; Haworth et al., 2008; Langdon et al., 2007; Langdon et al., 2012).

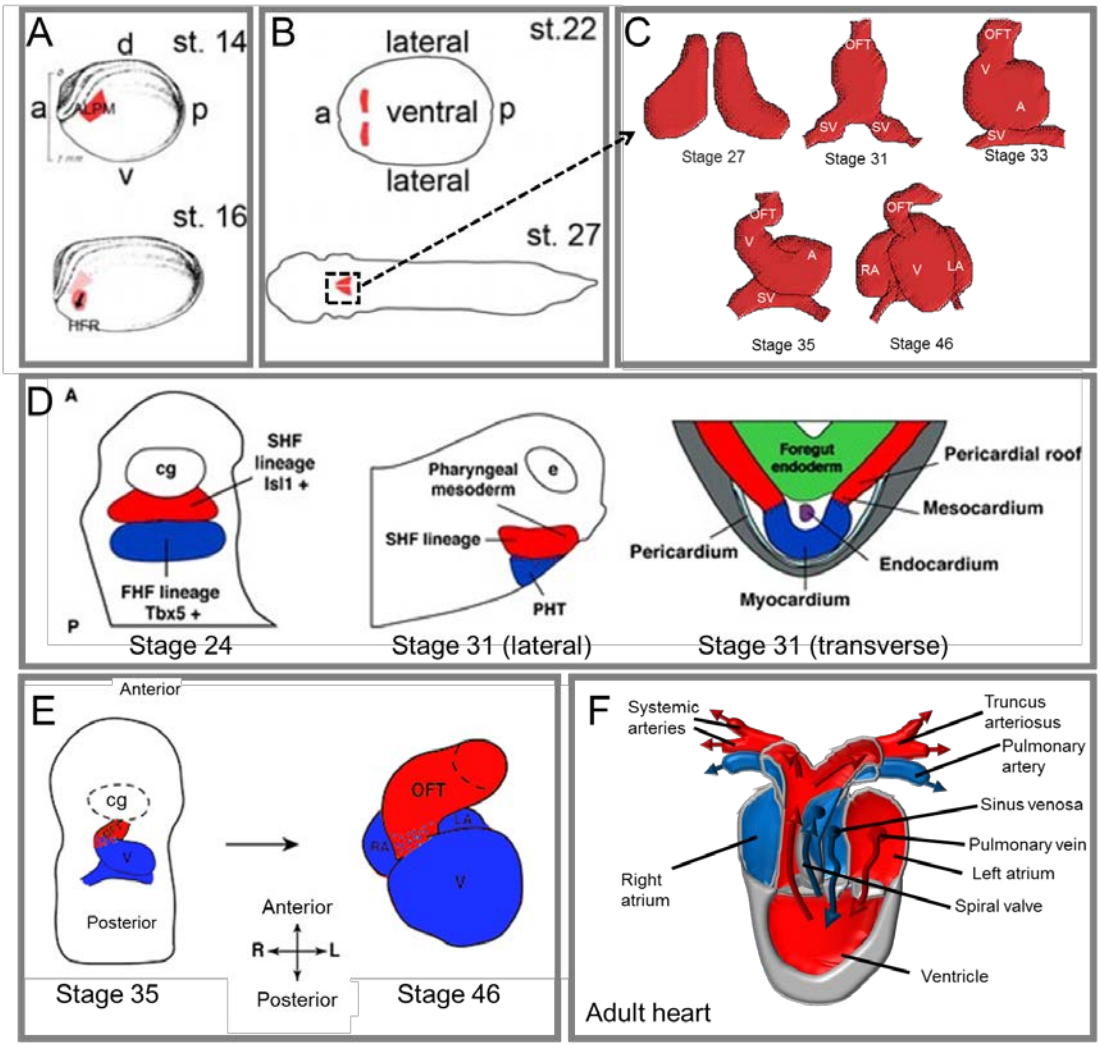

Figure 2. HPC movements to the ventral midline and later heart development.

Schematic of early heart development in (A) the neurula stage, (B) tailbud stages and (C) advanced stages of heart development. Panels A and B by Dr. Lance Davidson. (D-E) Relative contributions of first heart field (FHF; blue) and secondary heart field (SHF; red) to heart development. Panel D from (Pandur, Sirbu et al. 2013) and Panel E from (Hempel and Kuhl 2016). (F) Schematic of adult *Xenopus* heart showing oxygenated blood flow (red arrows) and deoxygenated blood flow (blue arrows), single ventricle and spiral valve, which functions to direct bloodflow to the lungs or systemic circulation.. ALPM = anterior lateral plate mesoderm, OFT = outflow tract, SV = sinus venosa, V = ventricle, A = atrium, R, = right, L = left, PHT = primary heart tube, cg = cement gland, e = eye.

The heart tube is three layers: an outer layer of myocardium, a mostly acellular cardiac jelly, and the endocardium. Endocardial precursors begin in the myocardial layer, before undergoing an EMT, migrating through the cardiac jelly before undergoing another MET contribute to the epithelial lining of the heart (von Gise and Pu 2012). The heart tube then begins to elongate and at the anterior end, the tube branches to form the truncus ateriosus, which later connects into the vasculature via the aortic arches (Fig. 2F). Myocardial contractions begin and as the tube continues to elongate, the outflow tract bends leftward while the conus begins a rightward looping movement (Stages 33-35; (Kolker, Tajchman et al. 2000)). Following looping, a septum forms between the atria (Stage 35-40) followed by myocardial wall thickening and trabeculation (Warkman and Krieg 2007). In amniotes, cardiac neural crest cells migrate in to septate the outflow tract, dividing pulmonary and system circulation (Waldo, Miyagawa-Tomita et al. 1998). This is not true for *Xenopus*, where instead cells of the secondary heart field play this role, with cardiac neural crest cells only contributing to the aortic arches (Lee and Saint-Jeannet 2011). At this point, the heart has three chambers, two atria and one ventricle and is actively pumping blood through the systemic vasculature (Fig. 2F).

1.2 BIOMECHANICS IN HEART DEVELOPMENT

The adult heart and vasculature can experience substantial remodeling and growth in response to mechanical stresses, so it should be no surprise that cardiogenesis in the embryo requires tightly regulated mechanical cues and properties to ensure proper development. The heart begins to function, i.e. pumping of blood, before its formation has completed, which

suggests these early mechanical behaviors may ultimately influence its own development (Bartman and Hove 2005). Here I review what is known about the mechanics that drive key points in heart development.

1.2.1 HPC movements to the ventral midline

During HPC movements to the HFR, these cells are moving in concert with the underlying endoderm. Microindentation studies combined with small molecule inhibitor treatments targeting actomyosin indicate the presence of a mediolateral stiffness gradient generated by actomyosin contractility within the HFR (Varner and Taber 2012). Performing microincisions to perturb the HFR and computational models to interpret the results, Varner and Taber determined that endodermal contractility is driving HPC movements into the HFR. This has later been confirmed in numerous zebrafish (Ye and Lin 2013, Xie, Ye et al. 2016) and chick studies (Aleksandrova, Czirók et al. 2012, Aleksandrova, Czirok et al. 2015). In zebrafish, it has been proposed that HPC and endoderm movements occur in distinct phases: co-migration of tissues, subduction, then independent migration of HPCs (Ye, Xie et al. 2015). These movements, driven by endoderm convergence, appear to be regulated by sphingosine-1-phosphate (Ye and Lin 2013) and YAP1 (Fukui, Terai et al. 2014), with the involvement of the latter heavily implying that there could be a mechano-feedback loop involved in this process (Dupont, Morsut et al. 2011).

Analysis into the migration trajectories of HPCs has shown that in chick, the movements of these cells are predominantly due to bulk convective movements of ventral tissues (Zamir, Czirók et al. 2006, Aleksandrova, Czirók et al. 2012). Indeed, autonomous migration of HPCs contributes very little to their ventral movement, although HPCs autonomously move anteriorly

relative to the endoderm (Aleksandrova, Czirok et al. 2015). By reducing endodermal convergent extension by performing microincisions to the ventral endoderm, Aleksandrova and colleagues were able to show that HPCs could not reach the ventral heart forming region alone despite moving some small distance ventrally.

The role of ECM contributing the environmental stiffness as well as cell mechanosensing is seen in numerous cases throughout development (Wozniak and Chen 2009, Lu, Takai et al. 2011, Schedin and Keely 2011). During HPC ventral movements, the ECM surrounding HPCs, composed primarily of fibronectin until fibrillin is also expressed at later stages, is also extensively remodeled (Aleksandrova, Czirok et al. 2015). Hand2, a transcription factor expressed in HPCs, regulates fibronectin deposition non-autonomously to ensure a proper environment for HPC merging on the ventral midline. Overexpressing Hand2 in HPCs leads to decreased fibronectin, cardia bifida and disrupted polarity among HPCs (Garavito-Aguilar, Riley et al. 2010). Given the close association of HPCs, the endoderm and the fibronectin-rich ECM at their interface, it is tempting to speculate that mechanical cues are transmitted across these ECM proteins to drive cardiogenesis. Supporting this hypothesis, HPCs also experience an enrichment of β1-integrin at the endoderm interface during this period, which can play a role in cell mechanosensory pathways (Schwartz and DeSimone 2008).

1.2.2 Heart tube formation

Once HPCs reach the ventral midline, the single layer of epithelized cardiomyocytes delaminates from the pericardial progenitors to fold up into a heart tube. The mechanics that drive this process remain poorly understood. It is theorized that asymmetric actomyosin contractility drives the bending of the myocardial sheet into a cylinder, to drive cardiomyocytes

to become wedge shaped (Taber 2001, Bartman and Hove 2005). However, unlike other examples of epithelial tube formation like neural tube closure which form via apical constriction (Andrew and Ewald 2010), heart tube folding would require basal actomyosin constriction as the apical surface of cardiomyocytes faces away from the presumptive lumen. During this time, cardiomyocytes also begin to exhibit polarized expression of the ion channel Na^+,K^+-ATPase. The purpose of this is not entirely known, although it is speculated to rebalance ions in these cells, since they soon begin spontaneous contractions, and facilitate fluid transport to expand the pericardial coelom (Linask 1992). It is also then possible that this expansion of the fluid-filled coelom in part contributes to pushing the cardiomyocytes to bend into a tube. This may also explain the origin of cardiac edemas as a result of abnormal epithelialization of HPCs, as disrupted organization of Na^+,K^+-ATPase channels may contribute to erroneous fluid transport.

While still a tubular structure, the embryonic heart begins to pump blood although the pulsatile flow characteristic of the adult heart is not observed. A number of mechanisms for heart tube pumping have been proposed, including a traveling contractile wave to drive a peristalsis-like pumping, as seen in the gastrointenstinal tract, or valveless suction pumping, e.g. impedence pumping (Santhanakrishnan and Miller 2011). However, the former is the more biologically realistic methods, and through computational modeling, Taber and colleagues were able to demonstrate how the heart tube could transition between peristalsis and pulsatility through the growth of the endocardial cushions (Taber, Zhang et al. 2007), which have often been speculated to be crucial for proper function of the heart tube (Fishman and Chien 1997).

1.2.3 Looping

As the heart tube elongates, it begins to bend into a c-shape (c-looping) and then an s-shape (s-looping). A number of hypotheses have been proposed to explain the mechanism behind this, including differential growth, differential cytoskeletal contractility, swelling of the cardiac jelly and active cell shape changes (Taber 2014). Although differential hyperplasic growth has not been supported by evidence, it has been shown that c-looping is in part driven through differential hypertropic growth of the myocardial cells (Soufan, van den Berg et al. 2006), with other mechanisms playing a secondary role (Taber 2014). Actomyosin contractility is also required for cardiac looping, as knockdown of actomyosin with small molecule inhibitors results in unlooped hearts (Noel, Verhoeven et al. 2013, Shi, Varner et al. 2015).

Recently, there has been renewed interest in a classic hypothesis that the heart tube twisting is the result of buckling under the compressive force of the pericardial coelom. Using physical models, the process of c-looping and s-looping was recreated by simulating heart tube growth and axial compression of the pericardial cavity with an elastic rod in a hemispherical cavity (Bayraktar and Manner 2014). As appealing as this hypothesis is, looping behavior appears intrinsic to the heart tube itself, as *ex vivo* heart tube isolates will undergo looping (Noel, Verhoeven et al. 2013), although it is unclear if they accomplish the level of structural complexity present *in vivo*. Interestingly, hemodynamics do not appear to influence looping, as blocking heartbeat and flow appears to have no effect on the process in chick (Remond, Fee et al. 2006), although conflicting information has been reported in zebrafish (Hove, Koster et al. 2003).

1.2.4 Septation, Trabeculation and Valvulogenesis

Although blood flow is not required for looping, it is required for later stages. The embryonic heart will remodel in response to pressure load, e.g. hemodynamics are required for endogenous ventricular cardiomyocyte hypertrophy and myofibril increases (Lin, Swinburne et al. 2012). Occluding blood flow through the embryonic heart also perturbs the processes of trabeculation, valve formation and chamber septum formation (Granados-Riveron and Brook 2012). Endocardial and endothelial cells are highly sensitive to shear stress, sensing hemodynamic forces through molecular pathways like integrins or via primary cilia (Hierck, Van der Heiden et al. 2008, Hierck, Van der Heiden et al. 2008), and consequentially, a number of congenital heart defects have been connected to ciliary defects. Prior to valve formation, there is oscillatory flow in the heart tube, i.e. retrograde flow following a contraction. Endothelial cells sense this retrograde shear stress and respond by upregulating Kruppel-like factor 2a in order to induce fibrous valve formation (Vermot, Forouhar et al. 2009). Furthermore, knocking down heart contractility, through manipulating expression of genes like troponin T, cardiac actin and atrial myosin heavy chain, results in an array of defects including malformed endocardial cushions as well as valve and septum defects (Granados-Riveron and Brook 2012).

1.3 GOALS AND SPECIFIC AIMS

Congenital heart defects (CHDs) are the leading cause of death in infants in the U.S. In addition to the emotional impact on afflicted families, CHD-related inpatient surgeries alone cost the medical industry over $2.2B each year. Noonan Syndrome (NS) is one of the most common

CHDs with an estimated 4000 babies born with NS annually, half of which are associated with point mutations in the protein phosphatase SHP-2, the most common being SHP-2 N308D. Although recent studies have highlighted the importance of mechanics driving organogenesis, the physical mechanisms that underlie early heart formation and the origin of CHDs remain poorly understood. The goal of this proposal is to elucidate how cellular and tissue mechanics drive the earliest stages of heart development, including a mesenchymal-to-epithelial transition (MET) of heart progenitor cells (HPCs). Bilateral fields of heart progenitor cells (HPCs) originate in the anterior lateral plate mesoderm (ALPM) at the end of gastrulation and undergo MET as they migrate and merge into a single population on the ventral midline to initiate heart formation. Failure of this migration or disruption of planar cell polarity can lead to cardia bifida, chamber malformation, hypertrophy, and other defects attributable to HPC disorganization. The model organism *Xenopus laevis* has striking genetic and anatomical similarities to the human heart, particularly in early development, and has advantages over other model organisms due to its external fertilization and development, relatively large size and amenability to microsurgery, biomechanical testing and molecular techniques such as antisense knockdown and overexpression. To understand the origin of CHDs we propose the following aims:

Aim 1. To what extent are HPCs actively migrating during their movement to the ventral midline?

Hypothesis: HPC movement to the ventral midline consists of two phases: (i) mesenchymal HPCs initially residing in the ALPM are passively carried toward the ventral midline by the endoderm, and (ii) HPCs undergo an MET and exert medially directed traction during collective cell migration.

Rationale: Previous studies have indicated a role for both the endoderm and HPCs themselves in their movement to the ventral midline. Preliminary data indicates these movements are biphasic.

Aim 2. When and how do mechanical signals instruct MET in HPCs?

Hypothesis: HPCs sense mechanical tension, which is temporally controlled, which induces MET and collective cell behavior changes that enable migration to the ventral midline.

Rationale: MET is instructed by mechanical cues in deep animal cap cells and the ventral endoderm generates a tension gradient as a result of undergoing CE.

Aim 3. Is Noonan Syndrome (NS) a mechanics-, PAR- or MET-related pathology?

Hypothesis: SHP-2 N308D results in improper heart formation at least in part due to perturbations to the temporal control of cell mechanics and HPC MET.

Rationale: SHP-2 regulates cell polarity (the PAR3/PAR6/aPKC complex) and cell contractility (ROCK). As temporal control of HPC MET requires strict regulation of polarity factors and mechanical cues from the microenvironment, SHP-2 N308D likely compromises the robustness of MET.

The technical innovation includes setting a precedent for using traction force microscopy to analyze the mechanical behaviors of complex tissues, which can be readily applied to understand other aspects of morphogenesis and broader cell biology. The conceptual innovation of this project lies within its ability to establish a novel connection between mechanics and MET which is not only crucial for heart development, but has broader applications to disease pathologies including cancer. Successful realization of these aims will develop a clear model for early

cardiac development that defines how gene-environment interactions can impact the risk of CHDs, permitting future studies to develop of morphogenetic stage-specific therapeutic interventions. Furthermore, as biomimetic approaches that recapitulate morphogenesis prove the most effective for tissue regeneration therapies, this project will provide invaluable insight into strategies for heart regeneration, how to control of tissue self-assembly for reliable heart tissue repair and how to biomanufacture cardiac tissue replacements.

In Chapter 2 of this dissertation, I will review the role of mechanical signals regulate MET in development and disease. In Chapter 3, I will address Aims 1 and 2 to reveal how the endogenous mechanical microenvironment induces MET in HPCs and how that influences HPC migratory behaviors. In Chapter 4, I will extend Aim 2 into an embryonic mesenchymal cell aggregate model to demonstrate how bulk tissue compliance regulates MET. And finally, in Chapter 5, I will concentrate on Aim 3 to show how NS-related mutations in SHP-2 override the native mechanosensory system of HPCs.

1.4 SIGNIFICANCE

Congenital heart defects (CHDs) occur in almost 1% of all live births affecting approximately 40,000 newborns each year in the U.S. (Hoffman 1995, Hoffman and Kaplan 2002) and is the leading cause of death in infants ((Nembhard, Wang et al. 2010); see Fig. 3 for more details on CHD incidence). Noonan Syndrome (NS), an autosomal dominant congenital disorder with an estimated frequency of between 1:1000 and 1:2500 live births (Sharland, Burch et al. 1992), is one of the most common causes of inherited CHDs. Patients with NS present facial dysmorphology and various heart defects, most commonly pulmonary stenosis and

hypertrophic cardiomyopathy (Sharland, Burch et al. 1992, Noonan 1994). Approximately 50% of all NS cases can be attributed to missense mutations in the protein tyrosine phosphatase SHP-2 (Tartaglia, Mehler et al. 2001) but the effects of these mutations on cellular events during heart formation are unclear.

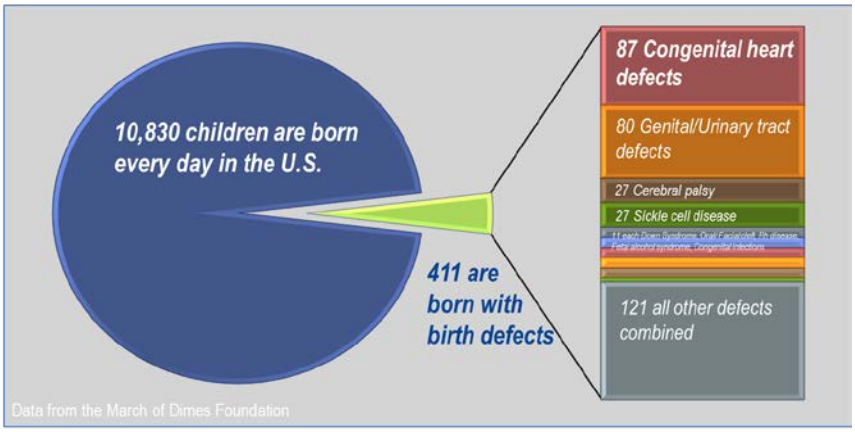

Figure 3. Incidence of congenital birth defects. Data from March of Dimes Foundation.

The importance of mechanics in organogenesis and the contribution of mechanical perturbations to birth defects have been recently highlighted (Nelson and Gleghorn 2012, Miller and Davidson 2013), yet, the physical mechanisms underlying early heart formation remain poorly understood. Recent genomic analyses have identified candidate proteins and genetic mutations associated with various CHDs (Pierpont, Basson et al. 2007) but a clear model for early cardiac development remains elusive, making it difficult to link lesions in molecular pathways to mechanistic changes. Without a clear model of the mechanics and gene-environment interactions that drive heart formation, we cannot effectively characterize and ultimately prevent and correct CHDs. The earliest stages of heart organogenesis are also the least studied, with scientific advances limited by complexities of molecular signaling pathways, the lack of

distinctive cardiac marker proteins early in development and physical barriers to visualizing and imaging HPCs prior to their arrival at the ventral midline; this last point is particularly true in certain development model organisms, such as zebrafish, mouse and chick, where the combination of neural tube closure, head fold formation and forebrain elongation completely obscure the heart progenitor cell (HPC) movements. The model organism *Xenopus laevis* provides a unique opportunity to combine intravital imaging techniques with *ex vivo* tissue isolates and in vitro cell aggregates to fully visualize HPC behaviors and test the role of the microenvironment.

Although the role of sequential EMTs has been extensively focused on in cardiogenesis (Markwald, Eisenberg et al. 1996, Thiery, Acloque et al. 2009, Kovacic, Mercader et al. 2012), the role of the presumptive mesenchymal-to-epithelial transition (MET) that occurs during HPC migration to the midline has been ignored. Preliminary data indicates that the MET that occurs in HPCs during their movement to the ventral midline is a critical step in early heart development. This project will provide insight into the process and regulation of that MET, which is not only critical to heart development, but also has broader implications to generalized MET and its role in cancer metastasis (Kang and Pantel 2013, Sosa, Bragado et al. 2014) and organogenesis of other organs, including the kidney (Dressler 2002, Gerlach and Wingert 2014).

2.0 ON THE ROLE OF MECHANICS IN DRIVING MESENCHYMAL-TO-EPITHELIAL TRANSITIONS

The following chapter is a published review on the role of mechanics in driving MET in both development and disease (Kim, Jackson et al. 2016). In this review, we discuss the evidence pointing to how mechanical cues initiate, propagate, and stabilize polarization during the process of MET and suggest that METs in early development, organogenesis and cancer progression all share a basic framework. We also note that there is a wide "spectrum" of METs, ranging from in unison epithelialization to single cell METs. Finally, we finish by noting that a comparative analysis of METs and their relation to EMTs needs to be a focus of future studies. This work was written in close collaboration with Hye Young Kim.

MET is an intrinsically mechanical process describing a multi-step progression where autonomous mesenchymal cells gradually become tightly linked, polarized epithelial cells (Fig. 4). METs are fundamental to a wide range of biological processes, including the evolution of multicellular organisms, generation of primary and secondary epithelia during development and organogenesis, and the progression of diseases including cancer. In these cases, there is an interplay between the establishment of cell polarity and the mechanics of neighboring cells and microenvironment. In this chapter, I highlight a spectrum of METs found in normal development as well as in pathological lesions, and provide insight into the critical role mechanics play at each

step. We define MET as an independent process, distinct from a reverse-EMT, and propose questions to further explore the cellular and physical mechanisms of MET.

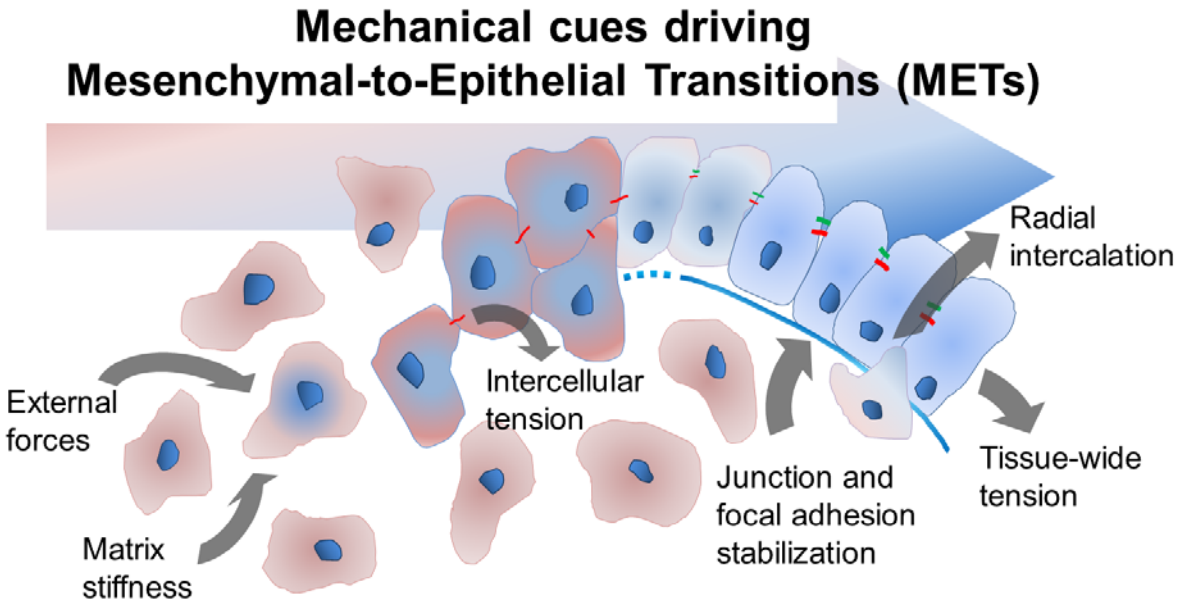

Figure 4. Summary of how mechanical cues regulate MET

2.1 INTRODUCTION

Mesenchymal-to-epithelial transition (MET) refers to the progressive phenotypic change from loosely associated motile cells to tightly bound cells with a distinct apical-basal polarity. MET is a fundamental cellular process that underlies a range of developmental, regenerative and pathological events relevant to human health. The emergence of MET in eukaryotes likely played a key role in the evolution of multicellular organisms from loose associations of single cells (Valentine 1978). The invention of specialized intercellular junctions, including tight and

adherens junctions, enabled cell-cell communication and allowed the creation of microenvironments isolated from the outside world in the form of lumens. As organisms became increasingly multifaceted in composition and physiological function, so did the programs of morphogenesis, which would require multiple sequences of EMT and MET. Cellular programs of MET from development eventually would be recapitulated during regeneration and under pathological conditions, driving formations of tumors and fibrotic lesions.

In this review, we will highlight extant cases of MET during development, examining the initiation and progression of METs in various contexts of self-assembling tissues, from early embryogenesis to metastatic tumor formations. We direct the interested reader to the excellent reviews have focused on the molecular signaling pathways that establish cell-cell junctions and apical-basal polarity (Kowalski, Rubin et al. 2003, Tsai, Donaher et al. 2012). Instead, we will focus on the specific steps of MET and the contribution of mechanical processes in driving MET, e.g. cell contractility, tensile forces and the mechanical properties of surrounding microenvironment. By relating the mechanical processes that drive mesenchymal cells to more polarized phenotypes we note many parallels with role of mechanics in regulating cellular processes including EMTs (Shook and Keller 2003, Gjorevski, Boghaert et al. 2012), cell survival and proliferation, and stem cell differentiation (McBeath, Pirone et al. 2004, Engler, Sen et al. 2006). In our conclusion we pose several questions to guide future research to elucidate how mechanical cues from the dynamically changing microenvironment influence the molecular signaling pathways driving polarity establishment.

2.2 DEFINING THE ROLE OF MECHANICS IN THE MESENCHYMAL-TO-EPITHELIAL TRANSITION (MET)

We broadly define a MET as any transition that increases the "epithelial-ness" of a mesenchymal cell, whereby the transitioning cell acquires a more polarized morphology or establishes a more asymmetric distribution of apical junctions. We include in our definitions events described as "epithelialization", "reversion to epithelia," and "re-epithelialization" but not these terms only span segments of a full multi-step MET (Fig. 5). Based on findings from a range of model systems that cover segments of the transitional process we propose multiple steps of MET that cover the entire cellular processes (Fig. 5) from (1) triggering the specification of epithelial cells via developmental programs or micro-environmental cues, (2) establishment of cell polarity, (3) propagation of MET through tissue, and (4) stabilization of new tissue architecture. Specific cases of MET and their mechanical regulation will be discussed in later sections.

2.2.1 Step 1: Initiation - the Decision to Change

The initial decision to transition from a mesenchymal to an epithelial phenotype can be categorized by the input signals, i.e., autonomous vs. non-autonomous. Of the many developmental METs found in development it is not clear how many occur autonomously; by contrast, numerous chemical or mechanical cues from microenvironment are known to drive MET, for instance as secondary metastatic tumors arise, or as iPSCs are generated from adult cells, or as wounds close.

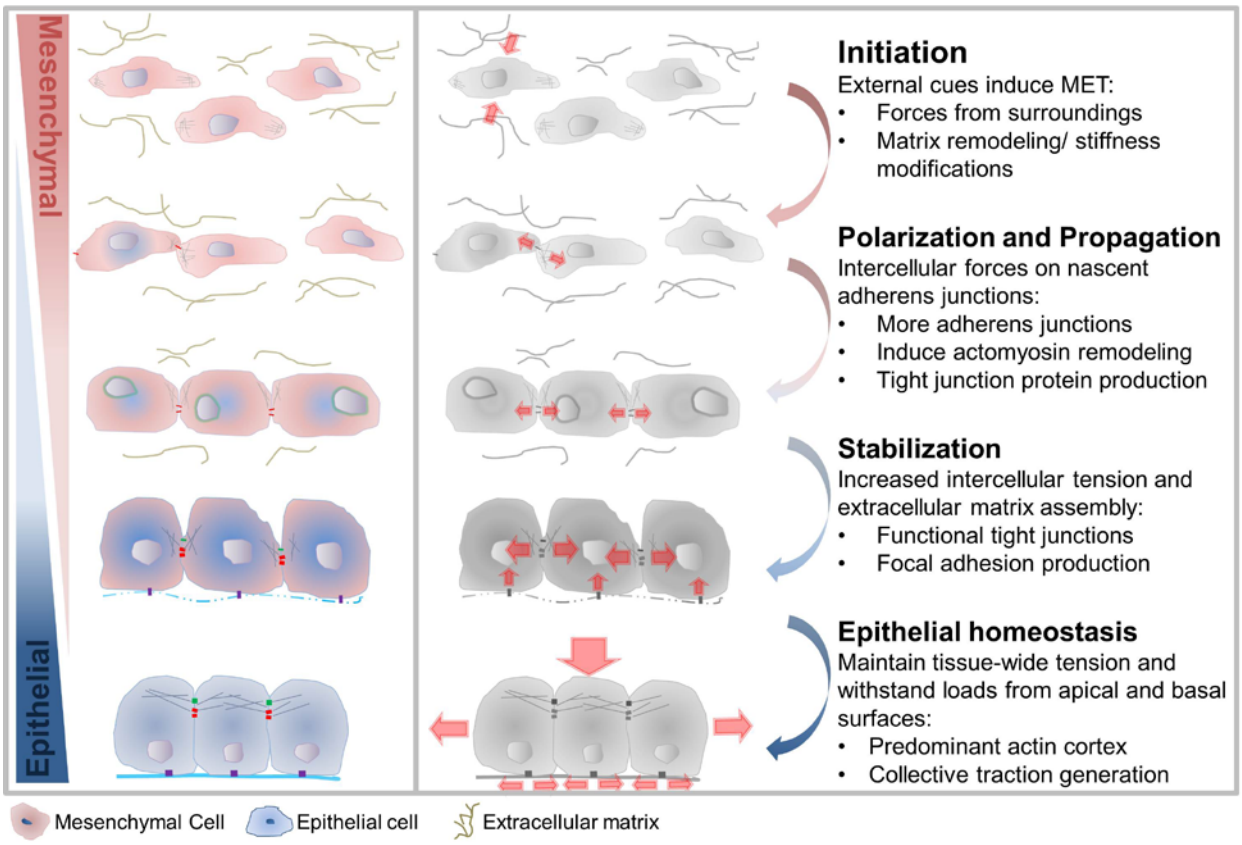

Figure 5. The role of mechanics in the step-wise progression of MET.

Scattered or loosely adhered mesenchymal cells (red) undergo multi-step progression to epithelial cells (blue): 1) Initiation, 2) polarization, 3) propagation, and 4) stabilization.

A number of cellular processes are known to be responsive to mechanical cues including stem cell fate decisions (Shibue and Weinberg 2011, Quail and Joyce 2013) and durotaxis in migratory cells (Ghajar, Peinado et al. 2013). A number of findings suggest mechanical cues contribute to MET; for instance, inner cell mass cells of the early mouse blastocyst undergo MET as they localize the polarity protein, aPKC, upon reaching the fluid-filled surface of the blastocyst cavity (Saiz, Grabarek et al. 2013). In another example, traction forces experienced by heart progenitor cells as they migrate ventrally in the embryo may trigger MET (personal communication, Tim Jackson). Such environmental cues may influence cancer cells, for instance

mechanical properties of the secondary site where circulating mesenchymal tumor cells reside is an important factor in activating their metastatic growth as epithelial tumor (Barkan, Kleinman et al. 2008, Barkan, El Touny et al. 2010, Samuel, Lopez et al. 2011, Cox, Bird et al. 2013).

2.2.2 Step 2: Polarization - Establishing a New Axis

After making the decision to adopt a more epithelial phenotype, mesenchymal cells need to establish apical-basal polarity. Cycles of actomyosin contractility drive the formation and maturation of cell-cell adhesion (e.g., E-cadherin; (Yamada and Nelson 2007)) between neighbors. Cells increase their adhesion to the ECM substrate by increasing numbers or increasing the strength of focal adhesions (e.g., integrin engagement through ECM and basement membrane;(Tepass, Tanentzapf et al. 2001)). Spatial patterns of junctional compliance, e.g. the "deformability" of cell-cell or cell-ECM attachments, localize assembly and activity of polarity proteins (e.g. Par3, Par6/aPKC, and crumbs; (Tepass, Tanentzapf et al. 2001)) that partition apical and basolateral membranes.

Throughout this process a thin meshwork of F-actin and myosin II under the cell cortex provide both mechanical stability and energy to remodel the cytoarchitecture. For example, soon after fertilization, the one cell embryo of *C. elegans* quickly clears the pulsatile actomyosin contraction from one side of embryo, stabilizing factors that establish anterior posterior polarity (Munro, Nance et al. 2004, Schonegg and Hyman 2006). This mechanically defined polarity translates into precise distribution of polarity-regulating factors (e.g., Par2 and Par6;(Munro, Nance et al. 2004)). The adhesion between E-cadherin expressing, MET undergoing cells, may nucleate actin polymerization and cortical contractility in neighboring cells. Cellular tension transmitted through the adherence junction can provide polarization cues to the rest of the cell

cortex and enhance the mechanical stability apical membranes. (Zhang, Betson et al. 2005, Yamada and Nelson 2007).

2.2.3 Step 3: Propagation - Spreading Polarity

There are many unanswered questions regarding the propagation of MET due to limited access to the real-time progression of MET *in vivo*. Several possible scenarios of MET propagation include (Fig. 6): 1) a single cell or a group of cells autonomously becomes epithelial until sufficient numbers form a functional epithelium, 2) a single cell or group of polarized epithelial cells recruit neighboring epithelial cells, through transmission of intercellular forces or signals, thus spreading a nascent epithelial sheet over the tissue, 3) apical insertion of cells from a basal layer of tissue to transition to epithelial. Cellular processes during propagation may be homologous to the processes that induce cell-cell junction formation in cultured epithelial cells after calcium depletion or wounding.

An intriguing possibility is that only a few cells undergoing MET may polarize their contractility and activate mechanoreceptors on neighboring cells that would lead to sequential MET induction along the axis of the tension. Alternatively, development of homogeneous tension over the surface of a compact group of mesenchymal cells or along the surface of a fluid-filled cavity might coordinate the onset MET at that surface.

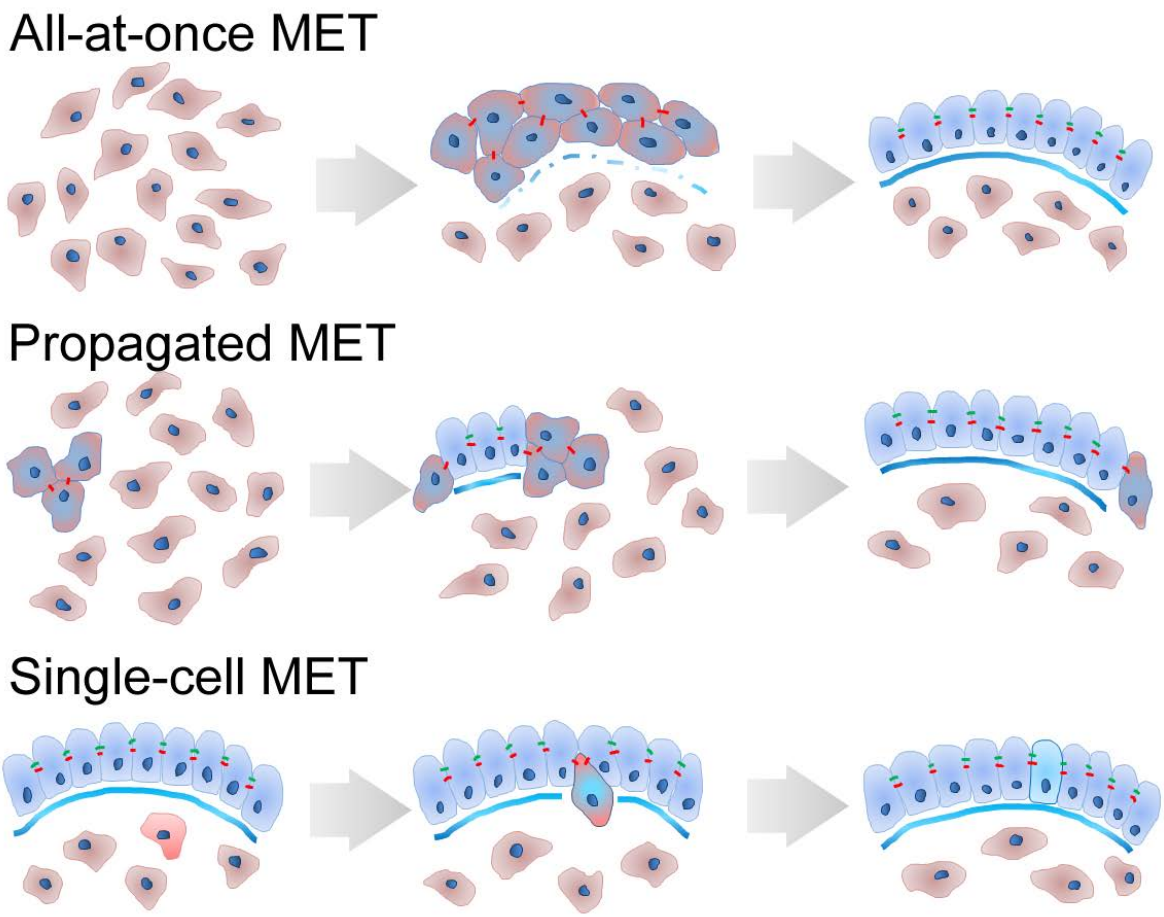

Figure 6. Cellular mechanisms of MET propagation.

Three strategies of MET that use mesenchymal cell sources (red) to expand and diversify an epithelial sheet (blue).

2.2.4 Step 4: Stabilization - Building New Architecture

Dense circumapical and apico-medial F-actin provide cells in the epithelium mechanical integrity and but also allow cell rearrangement and renewal throughout for the life of the organism. Tension within the epithelium, strength and cohesion of junctional complexes, and composition of the basal ECM may provide cues that guide for the insertion of specialized cells recruited from adjacent mesenchymal populations via single-cell MET (Chien, Keller et al. 2015, Sedzinski, Hannezo et al. 2016). Cycles of EMT and MET are common during regeneration of epithelial organs, e.g. after acute kidney injury, yet the role of mechanics in these events remains to be determined.

2.3 IN VITRO INSIGHTS TO METS

In vitro culture models have provided a valuable context to access and analyze the fine points of cellular mechanisms. Models of junction formation in stable epithelial cell lines and of junction re-establishment in cultured epithelial cells have been essential to identifying mechanisms that control junction formation and maturation, which offers partial insight into the steps of MET. In brief, currently available details of epithelialization (e.g., formation and establishment of adherens and tight junctions) are mostly explored using calcium switch protocols on cultured epithelial cells. Modulating simple factors including cell confluency and the period of calcium depletion have provided insight into various elements and magnitudes of re-epithelialization, including the temporal dynamics of localizing adherence junctions (E-

cadherin) and tight junctions (ZO-1)(Siliciano and Goodenough 1988, Adams, Chen et al. 1998), identifying the physical role of actin polymerization in sealing adherens 'zippers' (Kim, Pang et al.) and Rho-mediated contractile actin to strengthen epithelial junctions (Yamada and Nelson 2007). In addition to switching states of epithelial cells, mesenchymal cells (e.g., mouse fibroblast) expressing E-cadherin has been used to understand how cell polarity is established during MET and showed the role of cadherins in inducing epithelial-like polarization by restricting NaK-ATPase to the basolateral domains of the cell (Chaffer, Brennan et al. 2006).

2.4 METS AND EARLY DEVELOPMENT

Definitive stages of early development are synonymous with MET. Blastomeres localize their membrane traffic (Roberts, Leaf et al. 1992) and exhibit polarized membrane domains as early as the two-cell stage in the aquatic frog *Xenopus* (Chalmers, Strauss et al. 2003). Early mammalian morphogenesis begins with a MET when the 8-cell embryo undergoes compaction; undifferentiated cells become adherent and establish apical-basally polarized membrane domains (Fig. 7A) after activation of the apical-basal polarity pathways. The resulting polarity allows cells to apply contractile forces on neighbors via their adhesion sites (Winkel, Ferguson et al. 1990, Gumbiner 1996, De Vries, Evsikov et al. 2004, Plusa, Frankenberg et al. 2005, Fierro-Gonzalez, White et al. 2013). It is also thought that membranes surrounding early embryos, such as the vitelline membrane in *C. elegans* or the zona pellucida in mammals, exert a passive mechanical force on blastomeres that also contribute to compaction and constrain blastomere movements (De Vries, Evsikov et al. 2004, Grana, Cox et al. 2010). Upon division into the 16-

and 32-cell stages, only surface cells retain apical membrane polarity and the blastocyst cavity begins to form. In mammals, the surface cells and inner cells give rise to two different cell lineages, the trophoectoderm and the inner cell mass, respectively. Further development of the trophoectoderm requires tight junctions, actomyosin contractility, and cadherin-based adhesions (Fleming, McConnell et al. 1989), suggesting a continuing role for intercellular tension. Once inner cell mass forms apical tight junctions polarized distribution of NaK-ATPase pumps ions into the blastocoel which inflates from osmotic pressure (Roberts, Leaf et al. 1992). In zebrafish, cells forming the embryo are fully derived from an inner cell mass of mesenchymal cells and must undergo MET to establish all embryonic epithelia. The sorting and agglomeration of these cells appear regulated by differential contraction as cell-cortex tension regulates germ-layer specific epithelial aggregation (Krieg, Arboleda-Estudillo et al. 2008). Gastrulation proceeds with polarized cell orientations, intercalation and migration driving tissue collective migration and convergent extension (Jessen, Topczewski et al. 2002, Marlow, Topczewski et al. 2002, Ulrich, Concha et al. 2003, Yin, Kiskowski et al. 2008) requiring an interplay between cellular mechanics, canonical Wnt signaling and other planar cell polarity signaling pathways.

2.5 METS ARE FUNDAMENTAL TO VERTEBRATE ORGANOGENESIS

Mesenchymal cells contribute to the formation of many epithelia in the developing embryo through both multicellular and single-cell METs. In the following section we will discuss several examples that highlight specific mechanisms and different stages of MET during multicellular and single cell MET contributions to vertebrate organogenesis.

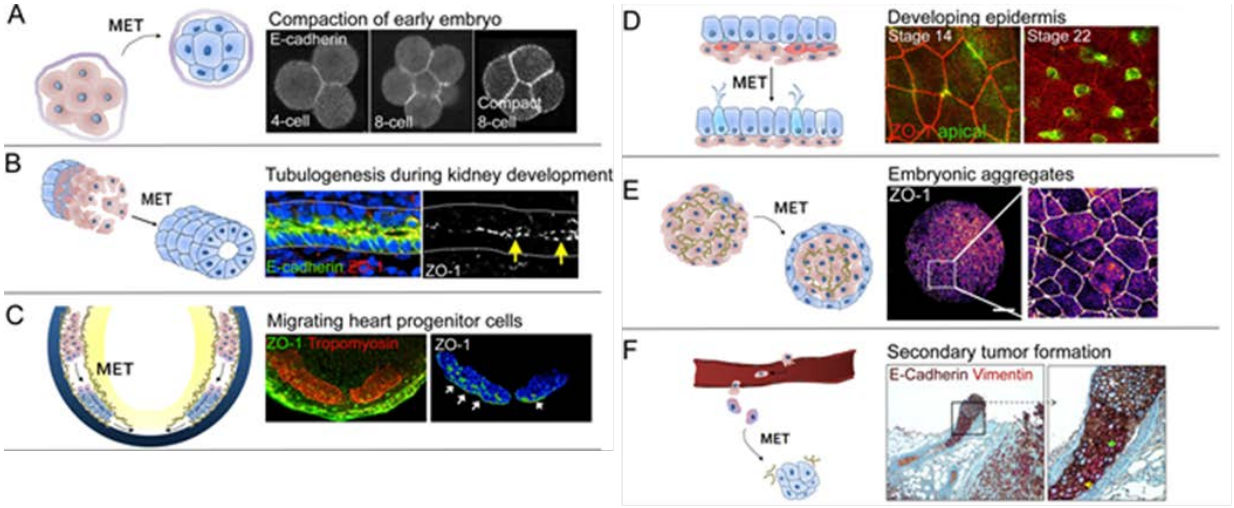

Figure 7. Six examples of MET during development, regeneration, and cancer.

(A) MET during compaction of the mouse embryo. Non-polarized egg initiates MET with E-cadherin mediated compaction of early embryo. Modified with permission from [105]. (B) MET during lumen formation of Wolffian duct (WD) in developing chicken embryo. Apical localization of E-cadherin and ZO-1 indicate formation of epithelia along the newly assembled lumen (yellow arrows). Modified with permission from (Attia, Schneider et al. 2015). (C) MET during heart development in Xenopus embryo (mid-tailbud; stage 28). Ventrally migrating heart progenitor cells (HPCs; red, tropomyosin) establish apical-basal polarity and form tight junctions (ZO-1, green on middle panel, pseudo-color in right panel - white arrows indicate apical surface; personal communication, T. R. Jackson). D) Single-cell MET during skin development of Xenopus embryo. Basal mesenchymal cells (green) intercalate and integrate with the pre-existing epithelial sheet (red ZO-1) (Stubbs, Davidson et al. 2006). E) MET in embryonic aggregates. Surface cells on pluripotent mesenchymal aggregate adopt apical-basal polarity and transition to epithelia cells (ZO-1). Pseudo-colored ZO-1 expression shows scattered clusters of epithelial cells across the surface of the mesenchymal-cell aggregate. F) MET in metastatic tumor. Heterogeneously mixed population of mesenchymal (vimentin) and epithelial (E-cadherin) cells found in tumor emboli within local lymphovascular space. Modified with permission from (Gunasinghe, Wells et al. 2012). Schematic diagrams in left panels (A-F) indicate involvement of mesenchymal cells (red) transitioning to epithelial cell type (blue).

2.5.1 Kidney Development

MET in kidney development is one of the best-studied METs in organogenesis and has provided detailed insights on the cellular basis and the molecular signaling pathways involved in kidney development, which have been well reviewed (Costantini and Kopan 2010). Kidney progenitor cells (KPCs) derive from bilateral fields of mesoderm cells that lie in between the somitic and lateral plate mesoderm. The caudally extending pronephric duct forms as mesenchymal leader and follower KPCs exhibit collective migration (Atsuta and Takahashi 2015, Attia, Schneider et al. 2015). Follower cells undergo MET to form a single-cell thick epithelial tube that extends caudally (Fig. 7B). Studies of collective cell migration models suggest polarized traction generation from the epithelial follower cells drive this process (Trepat, Wasserman et al. 2009), using intercellular tensile forces transmitted through cadherins to establish migration polarity (Weber, Bjerke et al. 2012). Such intercellular tensions may be promoting MET in the follower cells (Vasilyev, Liu et al. 2012) yet remain inhibited by FGF signaling in leader cells.

As the pronephric duct extends, adjacent mesenchymal cells form arrays of epithelial tubules that branch from the pronephric duct to lengthen medioventrally toward the aorta. The caudal end of the pronephric duct begins to swell to form the ureteric bud, which invades the metanephric mesenchyme and begins branching morphogenesis. At the tips of these branches, the metanephric mesenchyme cells undergo MET to contribute to branch elongation. Branching morphogenesis appears to be highly dependent on extracellular matrix (Santos and Nigam 1993), PCP signaling and convergent extension (Yates, Papakrivopoulou et al. 2010) and on Rho/ROCK and MAPK/ERK signaling, as inhibition of ROCK with Y27632 increases branching of epithelial buds and migration of metanephric mesenchyme (Meyer, Schwesinger et al. 2006).

Polarization of tubule epithelia appears crucial for kidney development, as perturbations give rise to defects, including polycystic kidney disease (Fischer, Legue et al. 2006). Fully-differentiated kidney epithelia include a diverse population of cell types; patterning of these intercalated cells depend on Notch signaling (Liu, Pathak et al. 2007) but how exactly they become incorporated into these epithelia, whether they present in the epithelium de novo or they arrive through single cell MET and intercalation, is not clear.

2.5.2 Early Heart Development

Vertebrate heart organogenesis is an inherently mechanical process whereby planar, bisymmetric fields of heart precursor cells (HPCs) move ventrally to merge on the midline, fold to form a lumen and tube, which loops and septates to form the adults heart. In avian and mammalian development, presumptive HPCs undergo an EMT during gastrulation and establish identities within the definitive mesoderm (Carver, Jiang et al. 2001, Shook and Keller 2003, Ohta, Suzuki et al. 2007, Thiery, Acloque et al. 2009) (Fig. 7C). Prior to reaching the midline, HPCs form epithelia that are characterized by basolateral membrane expression of B-catenin and apical aPKC (Trinh and Stainier 2004, Abu-Issa and Kirby 2007) and a belt of N-cadherin expression (Radice, Rayburn et al. 1997). Fibronectin is also deposited on the basal surface of HPC epithelia and plays a crucial role in establishing polarity and adherens junctions (Trinh and Stainier 2004). MET is required for proper heart function (Radice, Rayburn et al. 1997) and shortly after epithelialization is when the first cardiac electrical activity is recorded (Hirota, Kamino et al. 1987, Linask 1992). During their movement to the ventral midline, HPCs move through a dynamically changing mechanical micro-environment, which includes extensive fibronectin remodeling (Trinh, Yelon et al. 2005, Garavito-Aguilar, Riley et al. 2010) increases

in tissue stiffness (Zhou, Kim et al. 2010) and tension originating from the convergent extension of the endoderm (Varner and Taber 2012, Ye and Lin 2013, Shi, Varner et al. 2015). While knocking down actomyosin based cell contractility prior to heart tube formation and epithelialization consistently causes cardiac defects (Li, Zhou et al. 2004, Shi, Varner et al. 2015), perturbing cell contractility near the onset of looping, post-MET does not, which implicates a crucial role for cell mechanics in providing cues to guide the early organization of HPCs.

After the myocardial epithelium is established, endocardial cells undergo an EMT to migrate out of the trunk mesoderm and form a loose mesenchymal network located between the myocardium and the endoderm, which deposits and remodels fibrillin to enhance endocardial migration (Sugi and Markwald 1996). These cells aggregate to line the innermost lumen of the heart, and polarize their actin cytoskeleton, N-cadherin and integrin expression and NaK-ATPase (Linask 1992, Linask and Lash 1993). Additional trunk mesoderm cells contribute to the outflow tract and the right ventricle of the growing heart; disruption of this MET may have a role in congenital heart defects associated with DiGeorge Syndrome and basal filopodia activity appears crucial to outflow tract development (Francou, Saint-Michel et al. 2014). Both of these processes involve restructuring ECM, directed migration, and polarized actomyosin, suggesting that microenvironment mechanics or intercellular forces play a role in these METs.

2.5.3 Somitogenesis

After gastrulation the internalized presomitic mesoderm (PSM) is further subdivided and organized through a series of transitions that increase order within mesenchymal tissues

(Davidson, Keller et al. 2004, Latimer and Jessen 2010). New ECM interfaces provide cues to mesenchymal cells allowing them to polarize in shape and organize dynamic behaviors along the anterior-posterior and medial-lateral axes of the embryo (Barrios, Poole et al. 2003). Segmentation of the dorsal paraxial mesoderm into somites involves a further transition toward an epithelial phenotype. Avian exhibit the most polarized example of epithelial somites with clear apical-basal assembly of N-cadherin and ZO-1 demonstrating the somites in these animals generate an inward directed apical polarity (Duband, Dufour et al. 1987, Nakaya, Kuroda et al. 2004). The PSM in teleosts and amphibians, which undergo more rapid larval development, also develop highly ordered somites but these do not fully epithelialize with apical junctions (Fagotto and Gumbiner 1994, Daggett, Domingo et al. 2007). Interestingly, somitogenesis in these species appear sensitive to same factors that destabilize epithelial somites suggesting that their constituent mesenchyme undergo early phases of MET but do not complete the process.

We can consider the role of mechanics in somitogenesis MET in several stages. First, mechanical cues triggered by the somitogenesis clock may initiate phenotypic changes in cell behaviors and gene expression. Next, mechanical processes likely drive sorting as polarized mesoderm cells separate and engulf non-polarized cells. Lastly, forces produced at new surfaces of the PSM and nascent somite may stabilize junctions by feedback through intracellular trafficking pathways, using cues from surrounding ECM to orient assembly of apical-basal junctions. Patterns of cell-cell tension may enable cells to adjust their phenotype or even gene expression to their position within each new segment. Recent efforts to consider the impact of global patterns of tissue strain (Truskinovsky, Vitale et al. 2014) and the connection between cell behaviors and tissue mechanics during somitogeneis (Benazeraf, Francois et al. 2010) suggest mechanics may play an important role in the METs driving segmentation.

2.6 SINGLE-CELL MET - EXPANDING THE FUNCTION OF EXISTING EPITHELIA

Once an epithelial tissue or organ has been established, additional cases of single cell MET can generate specialized cells (Fig. 6). During single cell MET, individual cells within the surrounding mesenchyme are induced to become a specialized cell precursor which subsequently migrates from the mesenchyme and insert from the basal surface of the epithelium. The exact origins and the environmental cues generating most specialized cells remain unclear but we will review efforts to identify mechanisms underlying the contribution of single cell MET to the *Xenopus* embryonic skin and mammalian kidney tubules.

Single cell METs contribute distinct epidermal cell types responsible for ciliary transport and electrophysiology in the larval epidermis of the aquatic frog *Xenopus laevis* (Dubaissi, Rousseau et al. 2014) (Fig. 7D). Cells on the embryo surface differ from deeper mesenchymal cells by exhibiting strong apical-basal polarity both morphologically and molecularly (Jones and Woodland 1987, Chalmers, Welchman et al. 2002, Chalmers, Strauss et al. 2003, Chalmers, Lachani et al. 2006). Mesenchymal cells associate strongly with nascent fibronectin extracellular matrix, assembling a delineated basement membrane before gastrulation begins (Ramos and DeSimone 1996). Global movements of epiboly and gastrulation spread epidermal precursors within the embryonic ectoderm over the entire surface of the embryo where local signals mediated by Notch generate a set of precursor cells (Deblandre, Wettstein et al. 1999). Single cell METs in the *Xenopus* epidermis occur in stages as induced cells disperse along the basal surface of the existing epithelium and extend protrusions toward the apical surface of the epidermis (Stubbs, Davidson et al. 2006, Sedzinski, Hannezo et al. 2016). The cell mechanics of intercalation are poorly understood but appears to be regulated by many of the same pathways

that regulate cell motility and establishment of junctions between same-type epithelial cells (Mitchell, Jacobs et al. 2007, Mitchell, Stubbs et al. 2009, Sedzinski, Hannezo et al. 2016).

2.7 METS DURING DROSOPHILA DEVELOPMENT

METs play key roles in shaping the *Drosophila* embryo forming secondary epithelia such as the midgut and the dorsal vessel, adding diverse cell types to the renal rudiment, and later establishing the follicular epithelium of the developing oocyte. In contrast to vertebrates, junctions in epithelia descended from the blastoderm epithelia (e.g. primary epithelia) differ from junctions in midgut and heart tubes that form by MET from mesoderm cells produced by gastrulation (e.g. secondary epithelia; (Rugendorff, Younossi-Hartenstein et al. 1994, Tepass 1997)). *Drosophila* offers many lessons in morphogenesis and we summarize the role of MET in three events: formation of the heart tube, transformation of the mesoderm into the midgut epithelium, and intercalation of mesenchymal cells into the renal epithelium.

Formation of the cardiac heart tube, or dorsal vessel, requires transition of mesenchymal cells to a more epithelial phenotype with bilateral populations of cardiac precursors polarizing lumenal-domains before reaching the dorsal midline. Processes involved in MET and heart tube formation in *Drosophila* parallel those guiding vertebrate heart formation, including the dependence of MET, but not cell identity on extracellular matrix (Yarnitzky and Volk 1995, Chartier, Zaffran et al. 2002, Knox, Moyer et al. 2011, Harpaz, Ordan et al. 2013), and the role of tension within the embryo to position precursors and enable fusion at the midline (Haack, Schneider et al. 2014). Furthermore, amnioserosa cells, which first appear to engage precursors

with E-cadherin and septate junctions must disengage before cardiac cells can fully establish polarized lumenal-domains and fuse (Medioni, Astier et al. 2008). Mechanical cues within the microenvironment during dorsal closure and fusion are likely key to the MET and other phenotypic changes in cardiac precursors in the fly as they form the dorsal vessel.

Shortly after gastrulation mesodermal cells from the anterior and posterior midgut invaginations bridge the length of the embryo to establish the alimentary canal (Reuter, Grunewald et al. 1993). Mesenchyme progressively adopt apical-basal polarity with junctional E-cadherin and apical arrays of F-actin as they form a columnar epithelium. Just as in the case of the heart, extracellular matrix cues are not required for early patterning but are required for MET (Yarnitzky and Volk 1995). Similar to the dependence of MET and cell migration on laminin, in the forming midgut also depends on instructive cues from netrin expressed by their substrate through the midgut localized receptor frazzled to establish a columnar epithelium.

Once primary and secondary epithelial form in *Drosophila* they provide a substrate for single cell MET through radial intercalation. Intercalation of mesenchymal precursors into established epithelia, like intercalation of ciliary cells and ionocytes into the *Xenopus* epidermis, occurs in the midgut (adult midgut precursors and interstitial cell precursors; (Tepass and Hartenstein 1995) and renal epithelia.

2.8 MET REQUIRED FOR STEM CELL REPROGRAMING

MET is required during reprograming of somatic cells (Li, Liang et al. 2010, Samavarchi-Tehrani, Golipour et al. 2010). During the initial phase of reprograming, mouse embryonic fibroblasts make transition to an epithelial, unstable intermediate state through a MET(Takahashi

and Yamanaka 2006). This transition is marked by downregulation of mesenchymal genes (i.e., snail, N-Cadherin, fibronectin) and upregulation of epithelial genes (i.e., E-cadherin and Epcam). Recent studies highlight the role of mechanics in regulating MET during the generation of iPSCs; when cultured on microgrooves, induced fibroblasts align and elongate and show increased efficiency of iPSC formation which depends on MET (Downing, Soto et al. 2013). Failure to maintain cellular tension generated by the actin myosin network on microgrooves, by inhibiting myosin contractility with blebbistatin, completely abolishes cell reprograming and inhibits MET (Downing, Soto et al. 2013). Knock-down of these kinases that disrupt the F-actin network, including TESK1 and LIMK2 that phosphorylate the actin-binding protein cofilin (Sakurai, Talukdar et al. 2014), induce dramatic cellular phenotypic changes, turns fibroblasts into epithelial cells, and enhances iPSC efficiency. Similarly, MEFs cultured on a soft hydrogels increased MET and iPSC formation (Choi, Park et al. 2015). Together these data suggest a key role for cellular mechanics and tension maintained by cytoskeleton in regulating MET during iPSC formation.

2.9 METS IN SECONDARY TUMOR FORMATION

Metastatic cancers begins with dissemination of mesenchymal tumor which undergo MET and form proliferative macro-metastatic colonies. Comparing expression of epithelial junctional proteins including E-cadherin, β-catenin, and connexin, in primary tumor and matched distant metastases in lung, liver, and brain of cancer patients show equal or increased epithelial cells in metastases, indicating that circulating mesenchymal tumor cells undergo MET (Chao,

Wu et al. 2012, Atsuta and Takahashi 2015) (Fig.7F). Furthermore, activation of MET via repression of twist1 or prrx1 promotes cell proliferation and the establishment of metastatic colonies at distant sites (Ocaña, Córcoles et al. , Tsai, Donaher et al. 2012). Microenvironmental cues may inhibit or support MET and metastases (Shibue and Weinberg 2011); for example, a stiff microenvironment created by extensively crosslinked collagen fibrils has been shown to promote MET and secondary site establishment, tumor cell proliferation and metastatic colonization (Barkan, El Touny et al. 2010, Cox, Bird et al. 2013). Dense ECM fibrils at these sites can enhance integrin-mediated tumor cell engagement, activate focal adhesion kinase (FAK), and actomyosin reorganization (Barkan, Kleinman et al. 2008, Samuel, Lopez et al. 2011). Tension generated by the actomyosin network also provides feedback in the form of ECM arrangement and bulk tissue stiffness to drive tumor progression (Samuel, Lopez et al. 2011). The critical role of the mechanical microenvironment in cancer metastasis is increasingly appreciated (Nagelkerke, Bussink et al. 2015) but the role of these cues in driving MET are unclear.

2.10 CONCLUSION / FUTURE DIRECTIONS

The classical definition of MET relies on black and white definitions of what it means to be a mesenchymal cell or an epithelial cell relying on downregulation of mesenchymal markers, such as vimentin, with upregulation of epithelial markers, such as E-cadherin. This rigid, histology-based identification must be broadened in recognition of the diverse spectrum of functional METs observed in development and homeostasis. The number of studies that directly

contradict over-rigid definitions of epithelial and mesenchymal are increasing, including *Drosophila* mesenchymal cells requiring functional E-cadherin to migrate (Campbell and Casanova 2015), requirement of N-cadherin for the establishment and functional polarity of the outer epithelium in *Xenopus* embryos (Kintner 1992), functionally mesenchymal breast cancer cells that neither downregulate E-cadherin or express vimentin (Hiscox, Jiang et al. 2006, Hiscox, Morgan et al. 2006, Planas-Silva and Waltz 2007). Instead, METs come in various forms and degrees, ranging from a population of autonomous migratory cells that aggregate and form a structured epithelium or endothelium, to a single independent cell intercalating into an epithelium and adopting the polarity of the surrounding cells, to a cell sheet becoming slightly more polarized. In concluding this review we pose a number of key questions whose resolution will expand our understanding of MET and suggest avenues for improved approaches to regeneration and cancer treatment.

How do METs differ and what processes are conserved? While we have described a broad spectrum of METs that shape different tissues we propose they share common dependence on cell and tissue mechanics. Future studies will need to identify MET instructive cues and demonstrate how these regulators might be used to convert one type of MET to another. This strategy may prove crucial to elucidating key factors that regulate MET in disease progressions, including cancer metastasis, where direct observation is often logistically impractical due to the stochastic nature of MET in animal cancer models.

Is MET simply the reverse of an EMT? A number of cancer studies have identified what they call reversible EMTs, in response to growth factors (Gal, Sjoblom et al. 2008) and hormones (Planas-Silva and Waltz 2007). However, in development, many progenitor cell populations undergo a series of EMTs and METs that accompany dynamic changes to their

microenvironment. These transitions do not appear to be the “reverse” of each other, but rather a progression toward terminal differentiation. As studies uncover specific mechanisms driving MET, the contrast between MET and the so-called reverse EMT will become clearer.

What is the role of MET in regeneration? Mesenchymal-to-epithelial transitions appear to play a central role in regeneration of organs that formed initially via MET including heart and kidney and during regeneration of limb or tail structures where entire germ layers must be reconstructed. In these cases the precise role of MET in regeneration has been difficult to elucidate given the mixed lineage of organs and tissues comprised from multiple germ layers. For instance, kidney tubule regeneration in response to BMP7 has been proposed from renal fibroblasts (Zeisberg, Shah et al. 2005) but tracking the identity of the cells undergoing MET is challenging given the close lineage of the tubule endothelium and renal fibroblast. Still, efficient regeneration strategies may require induction and regulation of MET in order to recreate the diverse populations of cells needed for proper organ function.

3.0 PROPER HEART DEVELOPMENT AND FUNCTION REQUIRES TEMPORALLY CONTROLLED MECHANICAL CUES TO INDUCE MESENCHYMAL-TO-EPITHELIAL TRANSITION IN PROGENITOR CELLS.

The following chapter is a submitted manuscript. Here, I address Specific Aims 1 and 2 to investigate ventral tissue mechanics and how those mechanical cues control the timing of MET in HPCs. Certain elements of the manuscript, including the discussion and methods, have been expanded to provide further insight into my analysis methods and the interpretation of my results. All material that was originally located in the supplement has been moved into the main text.

During early cardiogenesis, bilateral fields of mesenchymal HPCs move from the anterior lateral plate mesoderm to the ventral midline undergoing an MET en route to form a single epithelial sheet. Through tracking of tissue level deformations in the heart forming region (HFR) as well as movement trajectories and traction generation of individual HPCs, we find the onset of MET correlates with a peak in mechanical stress within the HFR and changes in HPC migratory behaviors. Small molecule inhibitor treatments targeting actomyosin contractility reveal a temporally specific requirement of bulk tissue compliance to regulate heart development and MET. Targeting mutant constructs to modulate contractility and compliance in the underlying endoderm, we find MET in HPCs can be accelerated in response to microenvironmental stiffening and can be inhibited by softening. To test whether MET in HPCs was responsive to

purely physical mechanical cues, we mimicked a high stress state by injecting an inert oil droplet to generate high strain in the HFR, demonstrating that exogenously applied stress was sufficient to drive MET. MET-induced defects in anatomy result in defined functional lesions in the larval heart implicating mechanical signaling and MET in the etiology of congenital heart defects. From this integrated analysis of HPC polarity and mechanics, we propose that normal heart development requires bilateral HPCs to undergo a critical behavioral and phenotypic transition on their way to the ventral midline and that this transition is driven in response to the changing mechanical properties of their endoderm substrate.

3.1 INTRODUCTION

Physical mechanics are known to play a central role in shaping tissues, e.g. moving tissues into position, but their role in patterning cell behaviors and cell identity during development, growth, homeostasis, and disease is just now being explored (Paszek and Weaver 2004, Discher, Janmey et al. 2005, Wozniak and Chen 2009, Miller and Davidson 2013). Physical forces can induce or guide gene expression, polarity and differentiation (Engler, Sen et al. 2006, Mammoto and Ingber 2010, Weber, Bjerke et al. 2012). Likewise, cell invasion and behaviors during cancer progression and regeneration are known to depend on mechanical cues within the microenvironment (Barkan, El Touny et al. 2010, Samuel, Lopez et al. 2011, Nieto 2013). In particular, heart regeneration studies have provided evidence that both substrate stiffness (Kshitiz, Hubbi et al. 2012) and actomyosin contractility (Morikawa, Zhang et al. 2015) guide cardiac cell fates, suggesting that mechanical cues may play an important role in heart formation, since many of the same cell biological processes contribute to assembly of the heart.

Heart progenitor cell (HPC) identities are established following gastrulation in a bilateral population of mesenchymal cells within the definitive mesoderm (Shook and Keller 2003, Ohta, Suzuki et al. 2007, Thiery, Acloque et al. 2009). The two populations of HPCs then begin a large-scale movement from the anterior lateral plate mesoderm and undergo a mesenchymal-to-epithelial transition (MET; (Linask, Knudsen et al. 1997, Linask, Manisastry et al. 2005, Yelon and Stainier 2005)) before they merge on the ventral midline. In creating the architecture of the heart, HPCs undergo multiple transitions between mesenchymal and epithelial cell types (Markwald, Eisenberg et al. 1996, Thiery, Acloque et al. 2009, Kovacic, Mercader et al. 2012). Due to the precise timing of these transitions it has been proposed that small errors in the organization of the early heart field induce a cascade of defects leading to dramatic mutant phenotypes (Trinh and Stainier 2004, Abu-Issa 2014, Leung, Liu et al. 2015).

Classically, it was thought that HPCs actively migrate to the ventral midline (Trinh and Stainier 2004, Fukui, Terai et al. 2014); however, recently it has been suggested that the underlying endoderm, while providing substrate and growth factors (Nascone and Mercola 1995, Schneider and Mercola 2001), also plays a mechanical role in HPC movements (Varner and Taber 2012). Since HPCs move in concert with the endoderm, it has been proposed that ventral convergence and active contraction of the endoderm drive HPC ventral displacement (Varner and Taber 2012, Ye and Lin 2013) whereas autonomous HPC motility contributes minimally to their overall ventral movements (Aleksandrova, Czirók et al. 2012, Aleksandrova, Czirok et al. 2015, Ye, Xie et al. 2015). Thus, endodermal convergent extension is crucial for proper heart formation, as endoderm deficient embryos exhibit abnormal extracellular matrix and disorganized myocardial epithelia (Ye, Xie et al. 2015).

In addition to the involvement of endoderm, cardiac defects also arise from defects in early HPC polarity (Yelon and Stainier 2005, Ramsbottom, Sharma et al. 2014), actomyosin contractility (Langdon, Tandon et al. 2012, Ellawindy, Satoh et al. 2015, Shi, Varner et al. 2015), and the microenvironment of the heart forming region (HFR; (Kupperman, An et al. 2000, Garavito-Aguilar, Riley et al. 2010)). Even though these processes are likely involved in establishing the mechanical microenvironment of the HFR, the exact role of physical mechanics in early heart formation remains unclear. For instance, physical mechanical cues play a role in controlling cell phenotype during METs as well as epithelial-to-mesenchymal transitions (EMTs; (Kim, Jackson et al. 2016, Burute, Prioux et al. 2017)) and we have recently discovered a "mechanics-sensitive" case of MET in embryonic mesenchymal cell aggregates (Kim, Jackson et al. submitted).

To understand whether mechanics plays a role in the phenotypic transitions during heart formation we turned to investigate the mechanical microenvironment in the HFR and its role in guiding MET within HPCs as they move to the ventral midline. This phenotypic transition can be modulated by global, autonomous and non-cell autonomous changes in cell contractility and mechanical compliance. Furthermore, as HPCs undergo MET they change their migratory behaviors and start actively exerting traction. Lastly, we find temporal dysregulation of MET in ventrally converging HPCs leads to profound changes in heart anatomy and physiological function. These findings support a model whereby successful heart development requires strict temporal regulation of mechanics to drive MET in HPCs.

3.2 RESULTS

3.2.1 Heart progenitor cells undergo a mesenchymal-to-epithelial transition beginning at the mid tail-bud stage.

Studies in chicken and zebrafish have shown that HPCs undergo a MET and establish apicobasal polarity prior to their arrival at the midline (Linask, Knudsen et al. 1997, Trinh and Stainier 2004) but there was no description of apicobasal polarity in frog HPCs. To determine the timeline of MET progression in *Xenopus laevis* HPCs, we collected confocal transverse sections at three key stages of HPC movement to the ventral midline (Fig. 8A and B): the early neurula when HPCs are located within in the anterior lateral plate mesoderm (stage 15), the late tailbud when HPCs merge on the ventral midline and begin to express tropomyosin (stage 28), and an intermediate, mid tailbud stage (stage 23). HPCs appear mesenchymal at stage 15 (Fig. 8C) with little aPKC or ZO-1 (Fig. 1D-E). By stage 23, HPCs advance half-way to the ventral midline and apically localize aPKC (arrow in Fig. 8D). ZO-1 has not yet formed at apical cell-cell junctions but is instead found perinuclearly (arrows Fig. 8E), suggesting HPCs are preparing to form tight junctions (Gottardi, Arpin et al. 1996). Coincident with this transition in apicobasal polarity, HPCs become taller (Fig. 8C) and assemble fibrillin fibrils in the surrounding extracellular matrix (Fig. 8D). By stage 28, the bilateral heart fields merge and HPCs form two distinct single-cell layers, with their apical surfaces at the interface between these two layers. Columnar HPCs in the deeper, dorsal-most layer express tropomyosin (Fig. 8C; (Kolker, Tajchman et al. 2000)), and are strongly polarized with both aPKC and ZO-1 on their apical surface (arrows Fig. 8D-E). While fibronectin fibrils are localized to the ectoderm-mesoderm and mesoderm-endoderm interfaces at stages 15 and 23 (Fig. 8E), these fibrils decrease at the mesoderm-endoderm

interface by stage 28 where they are supplanted by dense fibrillin fibrils (Fig. 8D). At later stages, the deeper tropomyosin expressing layer delaminates from the more superficial layer and folds to form the tubular structure of the myocardium (Linask 2003). From these results, we conclude that stage 23 is the critical time point when MET is initiated in HPCs, approximately 7 hours before bilateral HPC fields merge at the ventral midline.

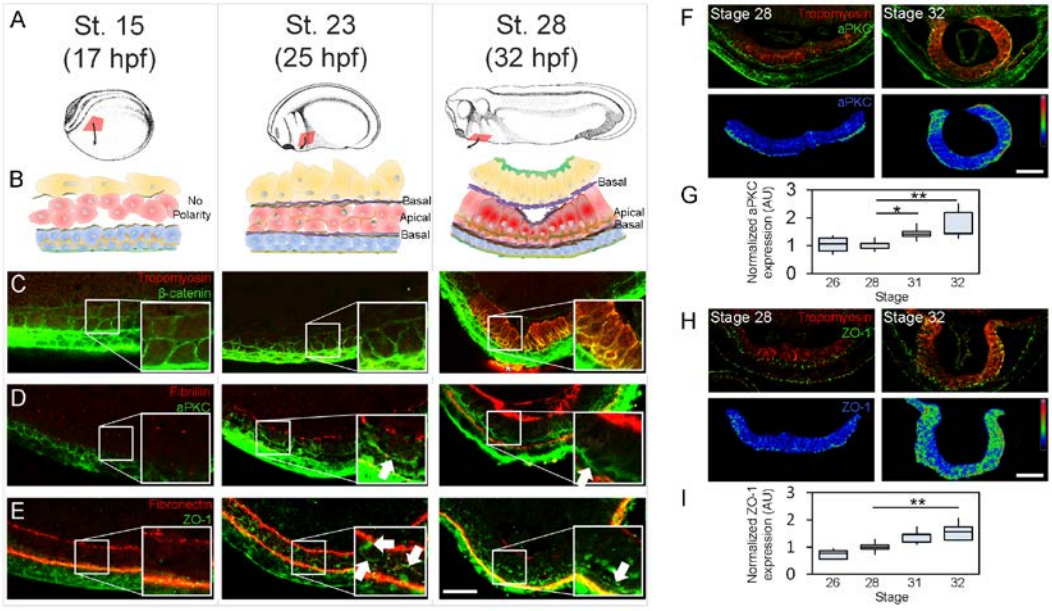

Figure 8. MET progression in *Xenopus* heart progenitor cells.

(A) Nieuwkoop and Faber drawings of *Xenopus* embryos at three stages during HPC movement from the anterior lateral plate mesoderm to ventral midline. HPC location shown in red with cell movement vectors overlaid. (B) Schematic of transverse section of HPCs and their microenvironment at the respective stage; blue = ectoderm, red = mesoderm, yellow = endoderm, brown = fibronectin, purple = fibrillin, orange = aPKC, green = ZO-1, dark red = tropomyosin. (C) Cell shape changes revealed by β-catenin (green) and cardiomyocyte marker tropomyosin (red). (D) Epithelial cell marker aPKC (green) appears on the apical surface of HPCs at stage 23 and fibrillin deposition (red) increases at the HPC basal surface over the stages investigated. (E) Tight junction protein ZO-1 (green) appears perinuclearly and nascently on the apical surface at stage 23 and appears strongly at HPC apical intercellular junctions at stage 28. Fibronectin (red) appears strong at the ectoderm-mesoderm interface all throughout these stages, but decreases at the mesoderm-endoderm interface at late stages. (F) Upper panels show transverse sections of stage 28 and stage 32 embryos stained for aPKC (green) and tropomyosin (red); lower panels show normalized aPKC intensity in pseudocolor LUT. (G) Normalized aPKC intensity prior to heart fields merging (stage 26), at time of merging (stage 28), during heart trough (stage 31) and heart tube formation (stage 32; N= 5-6 embryos per time point over 2 clutches). (H) Upper panels show transverse sections of stage 28 and stage 32 embryos stained for ZO-1 (green) and tropomyosin (red); lower panels show normalized ZO-1 intensity in pseudocolor LUT. (I) Normalized ZO-1 intensity over stages of heart development (N= 5-6 embryos per time point over 2 clutches). All scale bars are 50 µm. * indicates $p < 0.05$, ** indicates $p < 0.01$

The first signs of MET appear at stage 23 and apicobasal polarity continues to increase as HPC fields merge at the ventral midline and then fold to form a heart tube. Just prior to HPC fields merging, cardiomyocytes can be identified by their expression of tropomyosin (Kolker, Tajchman et al. 2000). Using the tropomyosin channel as a mask to identify cardiomyocytes, we quantified apicobasal polarity of HPCs by the intensity of aPKC and ZO-1 localization (Fig 8F-I). Intensity along the apical surface was measured and normalized compared to expression in the endoderm (Fig. 9). We see an approximately 50% increase in apical aPKC localization between the time the HPCs merge on the ventral midline (stage 28; Fig. 8F-G) and form a heart tube (stage 32; Fig. 8F-G). Similarly, apical ZO-1 localization also increases by approximately 50% over these stages (Fig. 8H-I). Apical epithelial markers first accumulate at stage 23 before HPCs reach the HFR; after stage 23 polarity progressively increases through stages of heart tube formation. Therefore, although we see the first signs of MET initiation at stage 23, apicobasal polarity progresses and propagates through stage 32.

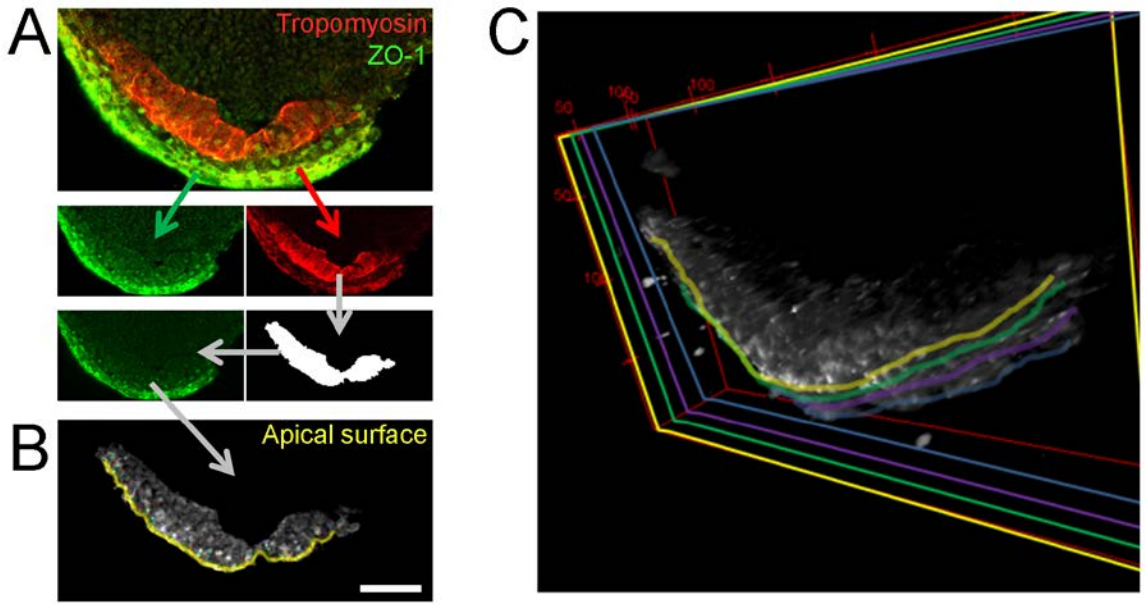

Figure 9. Schematic of epithelial marker apical intensity calculations.

(A) Transverse sections of stage 28 control embryo stained for tropomyosin (red) and ZO-1 (green). To calculate apical intensity, the red and green channels are first split. The tropomyosin channel is used to segment out and create a mask for cardiomyocytes. The ZO-1 channel has intensity normalized to background signal in the endoderm and undergoes a small Gaussian blur (kernel = 2 pixels). (B) Applying the mask to the normalized ZO-1 channel generates an image with only ZO-1 expression in cardiomyocytes. A 10-pixel wide linear region around along the apical surface is created to measure apical intensity. Scale bar = 50 μm. (C) 3D projection of the ventral, cardiomyocyte-masked, ZO-1 expression, showing apical intensity measurements at 4 different z-locations.

3.2.2 Perturbations to embryo bulk mechanical properties can delay or accelerate MET

Since transitions between mesenchymal and epithelial phenotypes can depend on mechanical cues (Shook and Keller 2003, Gjorevski, Boghaert et al. 2012, Kim, Jackson et al. 2016) and we have previously observed that changes in cell contractility and tissue compliance can accelerate or delay MET in *Xenopus* embryonic cells (Kim, Jackson et al. submitted), we

wanted to test whether MET in HPCs was similarly dependent. To perturb bulk tissue mechanics, we incubated embryos with small molecule inhibitors that target actomyosin contractility during the stages of HPC movement to the ventral midline (from stage 20 to 28, or approximately 11 hours) and observed whole embryo defects (Fig. 10A) including reduced anterioposterior (AP) length (Fig. 10B) and higher incidence of cardiac and neural edemas (arrows, Fig. 10A; Fig. 10C). To confirm the effects of small molecule inhibitors on ventral tissue mechanical properties, we measured the compliance (the inverse of stiffness) of the ventral HFR of a stage 23 embryo using microaspiration (Fig. 2D). Choice of a low pressure ($P < P_c$; (von Dassow, Strother et al. 2010)) for microaspiration reports the bulk, or composite tissue, compliance of a 125 μm diameter patch to reflect the contribution of all three germ layers in the HFR (Fig. 10D). These low suction pressures generate a small strain that mimics the physiological strain observed during development of the HFR. Most significantly, the myosin II inhibitor blebbistatin (Straight, Cheung et al. 2003) induced a two-fold increase in tissue compliance (J(120), Fig. 10E) whereas the myosin phosphatase inhibitor Calyculin A (Ishihara, Martin et al. 1989) and ROCK inhibitor Y27632 (Ishizaki, Uehata et al. 2000) produced a modest, although not statistically significant, 10 to 20% decrease and increase in tissue compliance, respectively. These findings implicate changes in tissue compliance as the cause of the observed cardiac defects.

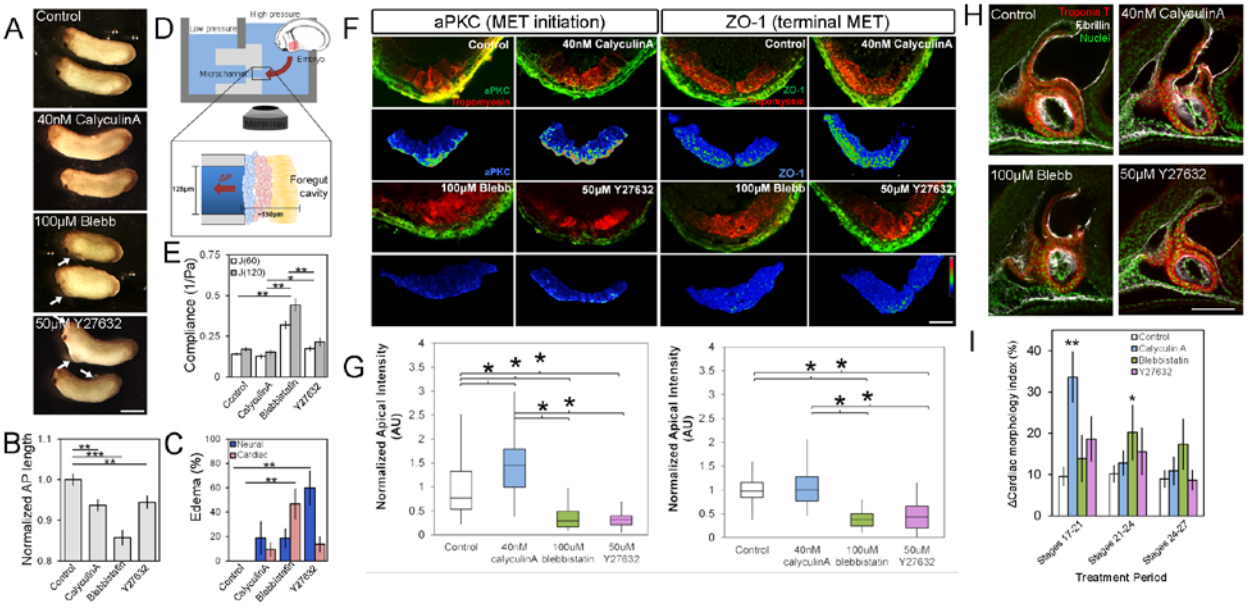

Figure 10. Modulating bulk tissue mechanics can accelerate or delay MET in heart progenitor cells.

(A) Embryos incubated in small molecule modulators of tissue compliance during stages of early heart development exhibit defects, including pericardial and neural edemas (see arrows). Scale bar = 1 mm. (B) Average anterioposterior length and (C) edema rate of small molecule inhibitor treated embryos. (D) Schematic of microaspriation showing region just lateral to the ventral midline is placed over the microchannel. Sketches after (Nieuwkoop and Faber 1967, Hausen and Riebesell 1991). (E) Microaspiration-based measurements of tissue compliance of the ventral tissue of stage 22 embryos confirms effects of blebbistatin and Y27632 on increasing tissue compliance and calyculinA on decreasing tissue compliance (N = 11-17 embryos per treatment over 3 clutches). (F) Transverse sections of the ventral midline at stage 28 of small molecule inhibitor treated embryos. Top panels show epithelial markers (aPKC or ZO-1) in green with tropomyosin (red) overlay. Lower panels show the epithelial marker masked using tropomyosin expression and with pseudocolor LUT to illustrate differences. Scale bar = 50 μm. (G) Calculations of the apical intensity of epithelial markers with respect to small molecule inhibitor treatment (N = 9-13 over 4 clutches). (H) Representative lateral view of immunofluorescence staining of stage 39 tadpole hearts; green = nuclei, red = troponin T, gray = fibrillin. Scale bar = 100 μm. (I) Quantitative analysis of cardiac anatomy with respect to stage-specific acute small molecule inhibitor treatments (N = 5 per treatment per period). Error bars represent mean ± SEM. * indicates $p < 0.05$; ** indicates $p < 0.01$.

To test whether tissue compliance plays a role in driving MET we assessed apicobasal polarity in HPCs by quantifying the apical intensity of aPKC and ZO-1 localization at stage 28, which is the earliest stage HPCs express tropomyosin (Fig. 10F). Calyculin A produced a 50% increase in aPKC and a modest increase in ZO-1 (Fig 10F-G). Y27632 and blebbistatin each produced a two-fold decrease in aPKC and ZO-1 (Fig 10F-G).To assess the effects of these treatments on late-stage heart development we washed out the inhibitors and cultured embryos until to the beating larval heart stage (+ 24 hours, stage 39). Small molecule modulators of tissue compliance and MET, particularly blebbistatin, produced abnormal cardiac anatomy and disrupted fibrillin organization (Fig 2H), indicating a malformed endocardium (Kolker, Tajchman et al. 2000). Importantly, none of the treatments prevented bilateral fields of HPCs from reaching the ventral midline or altered expression of cardiac markers (Fig. 10F). Taken together, these results demonstrate that perturbing mechanical properties of the HFR inhibit MET and induce later cardiac defects but do not overtly alter the movements of these cells to the midline or their differentiation.

Having established a timeline of MET in HPCs and the correlation between defects in MET and cardiac defects (Fig. 10), we wondered whether the stages of MET were sensitive to specific mechanical perturbations. We applied small molecule modulators of tissue mechanics for short time periods (Fig. 11A): 3 hours and 45 minutes either prior to MET (stage 17 to 21), during MET (stage 21 to 24) or after MET (stage 24 to 27). Following brief incubation, compounds were washed out and embryos were raised to stage 39 when heart morphology was assessed (Fig. 11B) based on size and shape of the larval heart (Fig. 11C). Cardiac morphology was highly sensitive to calyculin A prior to MET and blebbistatin during MET (Fig 10I) whereas modulating mechanics in the stages following MET did not significantly alter morphology. Since

heart development is most sensitive to increases in contractility prior to MET we propose that later defects in cardiac morphology are the result of precocious or accelerated MET. It is possible that calyculin A may hyperpolarize HPCs; however, the increase in apicobasal polarization we observe (Fig. 10G) remains within physiological levels seen at later stages (Fig. 8G). Thus, heart formation has stage-specific requirements for tissue compliance surrounding the onset of MET, suggesting that MET is dependent on temporally regulated mechanical cues.

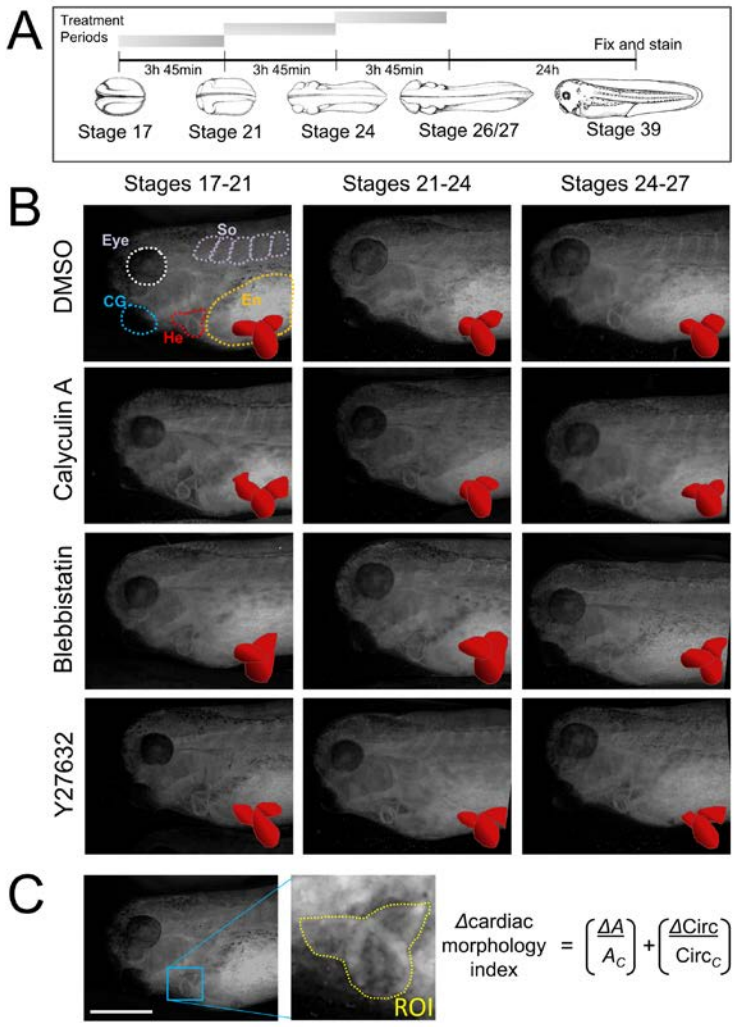

Figure 11. Acute small molecule inhibitor treatments reveal stage-specific requirements of actomyosin contractility for proper heart development.

(A) Schematic of different treatment periods and how they relate to embryonic development. Drawings after (Nieuwkoop and Faber 1967). (B) Stage 39 embryos exposed to acute stage-specific small molecule inhibitor treatments. Shape of heart drawn and expanded for emphasis (red). (C) The *Δ*cardiac morphology index is calculated from the morphology of a hand-drawn region of interest (yellow dashed line; ROI) surrounding the larval heart (calculated values in Fig. 2I). In addition to the heart (He) other features of the tadpole are visible including somites (So), cement gland (CG), and endoderm (En). Cardiac morphology (Δcardiac morphology index) is calculated from the difference of the heart area from mean (ΔA), average control heart area (A_c), difference of circularity of the ROI from the mean (ΔCirc), and average control heart circularity ($Circ_c$). Scale bar = 500 μm.

3.2.3 Heart progenitor cells undergo changes in behavior as they move to the ventral midline

Since cell migratory behaviors are thought to change as a result of MET (Dykxhoorn, Wu et al. 2009) and we observed a small number of cardiomyocytes excluded from the larval heart after actomyosin perturbations (Fig. 12, arrows), we tested the correlation between HPCs migratory behaviors and the timing of MET. At early neurula stages prior to MET, we observed mesenchymal behaviors in HPCs within HFR explants expressing moesin-GFP (Litman, Amieva et al. 2000) cultured with Cy5-tagged mAb 4H2, a non-function blocking antibody to visualize Xenopus fibronectin live (Davidson, Dzamba et al. 2008). At this early time, HPCs extend bipolar lamellipodia reminiscent of mesenchymal cells in dorsal mesoderm (Davidson, Marsden et al. 2006) as they migrate and deform the surrounding fibronectin extracellular matrix (Fig. 13). Since mesenchymal cell migratory behaviors are considered distinct from epithelial behaviors (Friedl and Gilmour 2009) we quantified HPC behaviors by following individual cell nuclei expressing H2B-mCherry in ventral "windowed" embryos and used nkx2.5-GFP expression to positively identify HPC tracks at the end of the time-lapse (Fig 13). We calculated migration parameters for each nucleus over three different 4-hour time periods: before MET (1 to 5 h), just following MET (7 to 11 h) and as heart fields merge (13 to 17 h). HPCs migration changes in both speed and direction in the time prior to MET (fast, anterior directed) compared to the later times (slow, posterior directed; Fig. 13C and D). Surprisingly, migration speed, persistence and collectiveness are all lower after MET (Fig 13E), contrary to more conventional examples of epithelial cell migration (Friedl and Gilmour 2009).

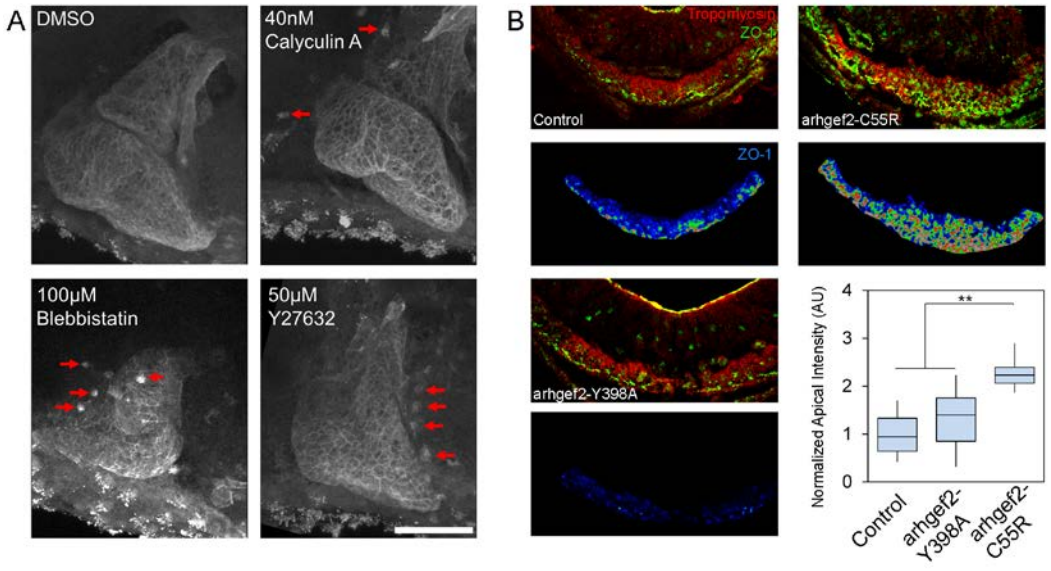

Figure 12. Chronic small molecule inhibitor treatment results in failure of cardiomyocytes to incorporate into vertebrate heart while targeted injections to perturb endoderm contractility with arhgef2 enhances ZO-1 polarity.

(A) 3D projections of Stage 39 embryonic hearts following chronic small molecule inhibitor treatment (stages 20 to 28, ~11 hours) stained for tropomyosin. Arrows show tropomyosin positive cells in the ventral heart forming region that do not assemble into the developing heart. Scale bar = 100 μm. (B) Transverse sections of stage 28 embryos stained for tropomyosin (red) and ZO-1 (green) with ZO-1 intensities masked by tropomyosin expression (pseudo-color). Embryos with endoderm injected with arhgef2-C55R show elevated ZO-1 expression relative to the rhodamine dextran (RDA) injected control and embryos injected with arhgef2-Y398A. Normalized intensities of arhgef2-C55R injected embryos are significantly higher than either RDA injected controls of arhgef2-Y398 injected embryos. Scale bar = 50 μm. ** denotes $p < 0.01$.

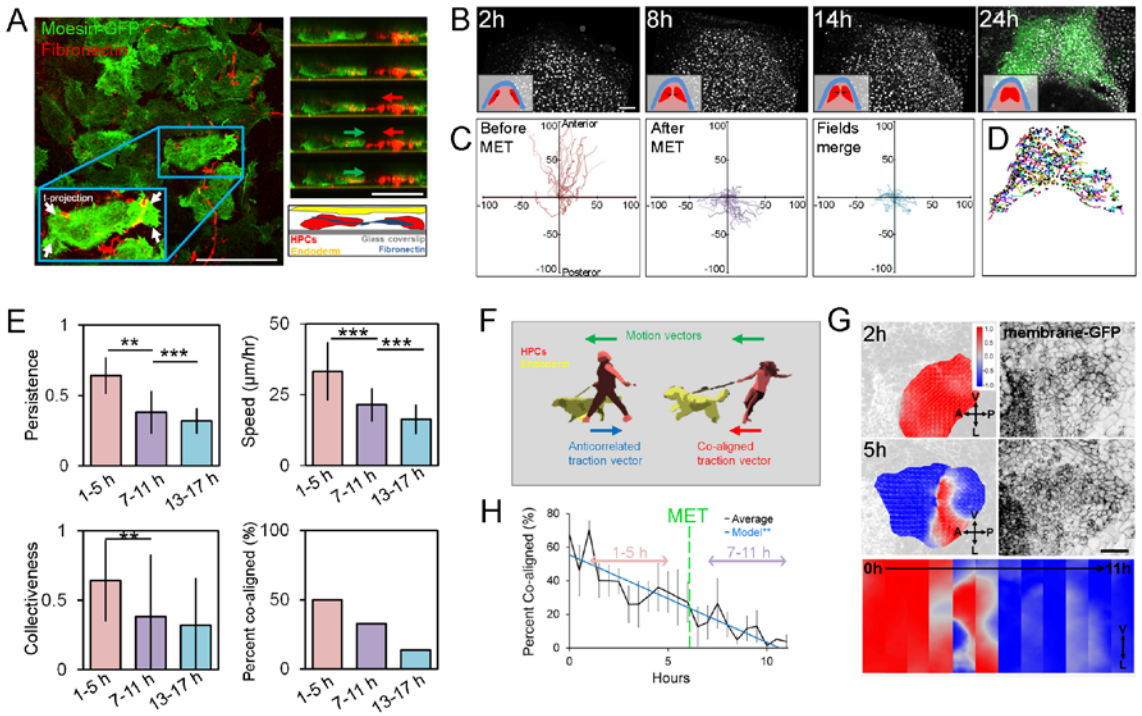

Figure 13. Heart progenitor cells undergo behavioral transition during stages of MET

(A) Left panel: heart progenitor cells isolated from the anterior lateral plate mesoderm show mediolaterally oriented bipolar lamellopodial protrustions (see arrows) that deform fibronectin fibrils (left). Right panel: HPCs extend protrusion, pull on fibronectin fibril (red arrow), and move forward (green arrow). (B) Intravital windowed embryos with fluorescently labeled nuclei reveal cell movements (inset shows tissue organization of ectoderm (blue), mesoderm (pink), and HPCs (red)). At 24 h, expression of nkx2.5-GFP transgene allows retrospective identification of HPC tracks (scale bar = 100 μm). (C) Spider diagrams of nuclei migration trajectories over 4 hour period. (D) Nkx-2.5-GFP identified HPC movements from start to end (black circle). (E) Persistence, speed and collectiveness of HPC movements before MET (1 to 5 h) are higher than after MET (7 to 11 and 13 to 17 h). N = 252 cells for 1 to 5 h, 216 for 7 to 11 h, 411 for 13 to 17 h pooled from 3 embryos from 3 different clutches. * indicates $p < 0.05$, ** indicates $p < 0.01$, *** indicates $p < 0.001$. (F) Schematic of traction force and motion correlation. (G) Correlation score between motion and bead displacement vectors on the left and membrane-MKO2 on the right. HPCs exhibit co-aligned motion and traction (+1, red) before transitioning to anticorrelated (-1, blue) (scale bar, 100 μm). Strip kymograph over 12 hour period (lower panel). (H) Percent of co-aligned HPCs decreases over MET (7 tissue isolates over 4 clutches, black; linear regression, blue; error bars, ± SEM).

Several studies have proposed that endoderm carries HPCs ventrally (Varner and Taber 2012, Ye and Lin 2013, Xie, Ye et al. 2016). To investigate the mechanical coupling between migratory HPCs and their microenvironment we used traction-force microscopy (TFM). By tracking movements of cells and their deformable 2D extracellular matrix substrate, TFM has been used to estimate traction forces generated by single cells (Sabass, Gardel et al. 2008), cell pairs (Maruthamuthu, Sabass et al. 2011) and cell monolayers (Tambe, Croutelle et al. 2013, Vincent, Bazellieres et al. 2015), but has not been widely applied to study traction forces generated by primary tissues or tissue explants (Dzamba, Jakab et al. 2009, Zhou, Kim et al. 2010). To evaluate coupling of HPCs to their local substrates we adopted a "low-resolution" version of TFM to detect changes in the orientation of traction forces, e.g. traction vectors, rather than absolute force magnitudes. If HPC movements were dominated by active migration their traction vectors would be anti-correlated with their motion vectors; conversely, if HPCs were carried by endodermal movements, their traction vectors would be co-aligned with their motion vectors (Fig. 3F). We found HPCs movement and tractions are co-aligned before MET (Fig. 13G, red) and transition to anti-correlated after MET (Fig. 13G, blue). By the time MET initiates (Fig. 13H, green line), only 25% of HPCs are co-aligned. Thus, as MET progresses, HPCs become less dependent on the underlying endoderm for movement and instead actively direct traction forces to aid their movement toward the ventral midline; consequentially, both MET and coupling to the endoderm may contribute to the reduced migration speed and collectiveness of HPC movements.

3.2.4 Ventral elongation rates and bulk tissue stiffness peak at stage 23

During the stages of heart progenitor cell movement to the ventral midline, dorsal and ventral tissues elongate nearly four-fold (Larkin and Danilchik 1999), suggesting that the mechanical microenvironment surrounding HPCs is highly dynamic. To determine whether mechanical changes during ventral elongation coincide with the timing of HPC MET we tracked tissue movements of the embryo's ventral anterior surface from late neurula stage (stage 17) using a custom image analysis program (Kim, Hazar et al. 2014)(Fig. 14A). Since HPCs are located within 50 μm of the ventral ectoderm surface of the embryo, and ectoderm and mesoderm appear mechanically coupled at these stages, our analysis assumes that deformation of HPCs and the HFR could be approximated by deformations observed in ventral ectoderm. Rather than elongate anteroposteriorly at a constant rate like dorsal tissues, the ventral heart forming region elongates in a complex nonlinear manner (Fig. 14B), initially shortening and then elongating (Fig. 14C). From deformations observed in time-lapse sequences we calculated an instantaneous strain rate, which represents the percentage elongation per hour. Strain rates reach a peak of approximately 11 % per hour at stage 23, which coincides with MET onset, before dropping to low, but steady, rates of elongation consistent with larval growth (Fig. 14D).

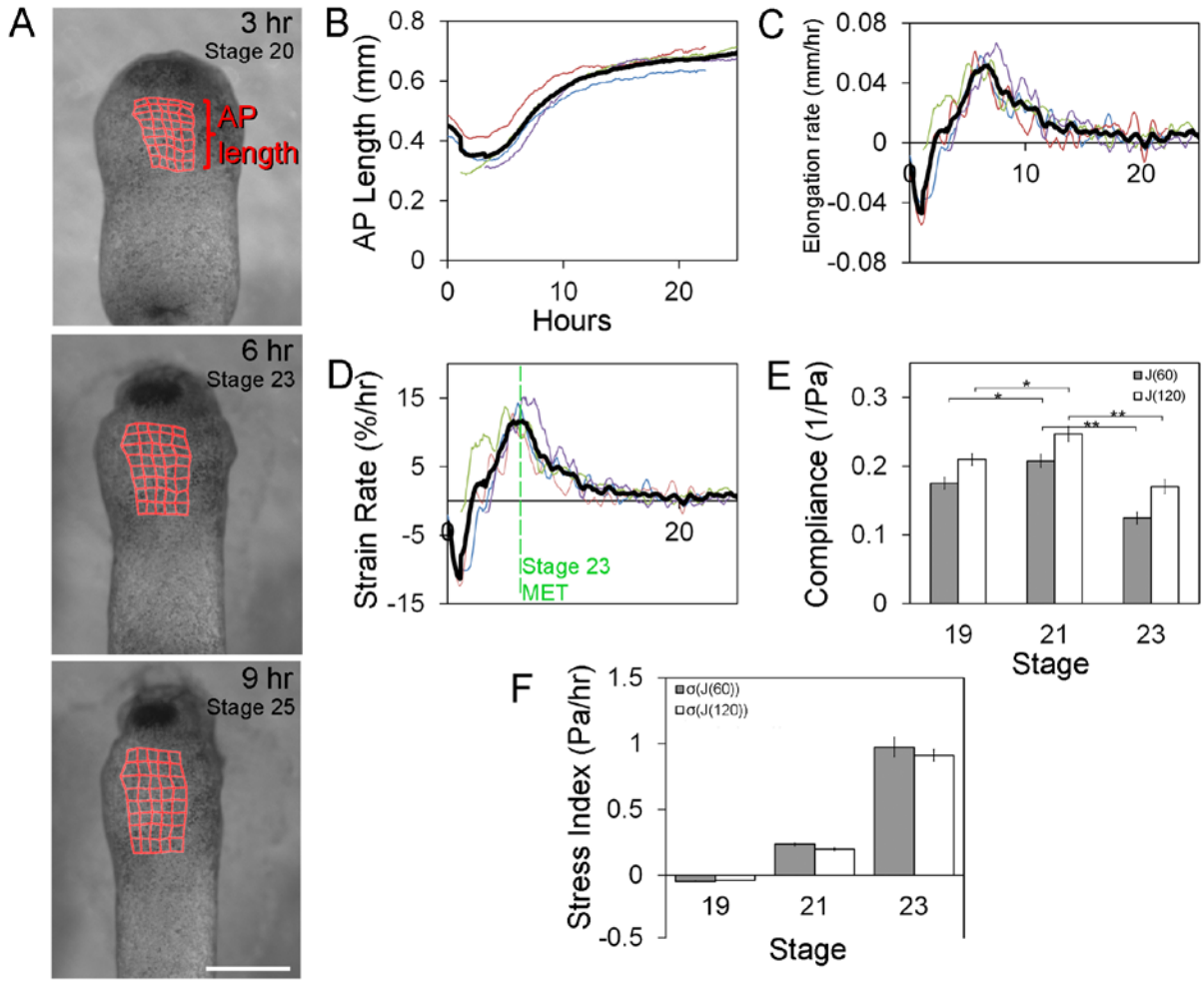

Figure 14. Ventral heart forming region experiences peak anterioposterior stress at stage 23.

(A) Ventral regions of stage 17 embryos were imaged for 24 hours and tissue deformations (red grids) were tracked automatically using custom software. Frames are shown at 3, 6 and 9 hours after the embryo was mounted. Scale bar = 500 μm. (B) Anterioposterior (AP) length tracked over time shows non-linear growth. AP length was used to calculate the instantaneous elongation rate (C) and the strain rate (D) over time. Graphs report data from four embryos, with the thick black line indicating the average. (E) Compliance data at three different stages, including fast-response (J(60)) and steady-state (J(120)) compliance. (F) Anterioposterior stress index (Pa/hr) at three different stages, calculated from compliance and strain rate data. Error bars represent mean ± SEM. * indicates $p < 0.05$; ** indicates $p < 0.01$.

In order to estimate stress experienced by HPCs we measured the mechanical compliance of the HFR with microaspiration. Compliance initially increases between stages 19 and 21 only to drop almost two-fold at stage 23 (Fig 14E). Combining compliance and strain rates, we see a sharp peak in mechanical stress experienced by HPCs at stage 23 (Fig. 14F). Thus, MET coincides with peak stiffness and strain rates in the HFR, suggesting that MET may be triggered in HPCs in response to high levels of mechanical stress.

3.2.5 Targeted injections to modulate endoderm contractility can accelerate or inhibit MET

Given the role of endoderm mechanics in driving HPC movements in chicken embryos (Varner and Taber 2012) and the dynamic changes in stress within the HFR in frog (Fig. 14F) we wondered whether HPCs were responding directly to mechanical cues from the endoderm within the HFR. To test this we reduced compliance specifically in the anterior endoderm by modulating actomyosin contractility and tested whether MET was accelerated in HPCs. To modulate actomyosin contractility in the endoderm we used the 32-cell *Xenopus* fate map (Dale and Slack 1987, Bauer, Huang et al. 1994) and injected dorsal vegetal blastomeres targeting exogenous mRNA to either HPCs or cells in the anterior endoderm (Fig. 15A and B). To reduce the compliance of endoderm, blastomeres were injected with constitutively active arhgef2-C55R, which we have previously shown to increase dorsal tissue stiffness two-fold (Zhou, Kim et al. 2010). Embryos expressing arhgef2-C55R in the anterior endoderm exhibited reduced AP lengths (Fig. 15C) and cardiac edemas (Fig. 15D), but no defects in cardiac tissue differentiation (nkx2.5-GFP, Fig 15B see arrows). To confirm the mechanical effects of arhgef2-C55R on ventral tissue mechanical properties, we microaspirated the HFR at stage 23 and found a 50%

decrease in tissue compliance (Fig 15E). At stage 28, we found a 50% increase in MET by aPKC localization (Fig 15F) and a nearly two-fold increase in ZO-1 localization (Fig. 12). The ROCK inhibitor Y27632, which operates downstream of arhgef2-C55R, was able to block accelerated MET with a two-fold decrease in aPKC localization (Fig. 15F). By stage 39, tadpoles with arhgef2-C55R expressing endoderm displayed abnormal cardiac architecture, including smaller chambers, perturbed fibrillin structure and hypertrophy (Fig. 15G), demonstrating that not only does induced endoderm contractility perturb MET, but that it is sufficient to generate cardiac defects.

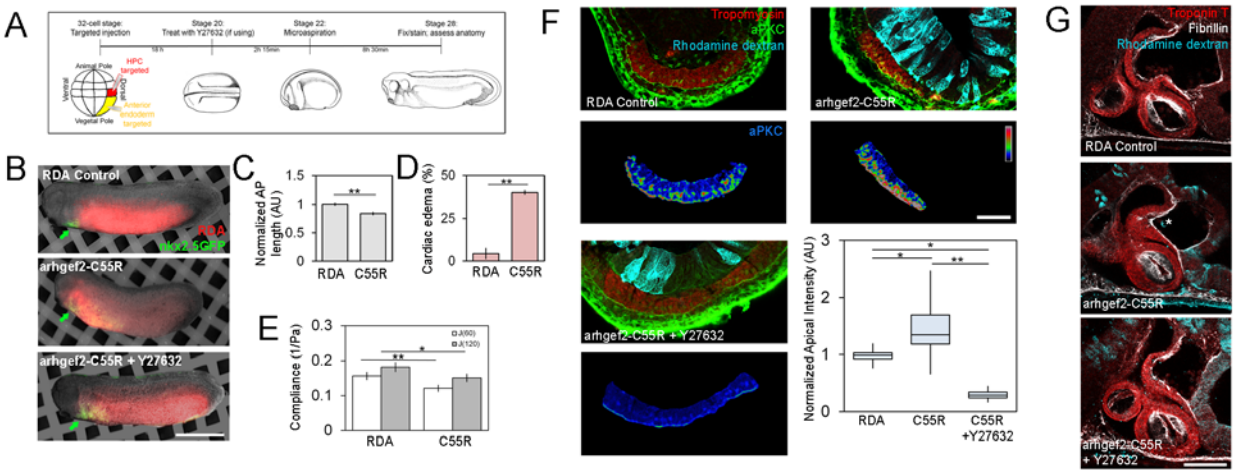

Figure 15. Targeted injections to perturb microenvironmental stress via actomyosin contractility accelerate MET.

(A) Experiment schematic and timeline. Injections took place at 32-cell stage to target either HPCs (red) or the anterior endoderm (yellow). (B) Transgenic nkx2.5-GFP (green) embryos co-injected with arhgef-C55R mRNA and rhodamine dextran (RDA; red) targeted to the endoderm. Cardiac differentiation is uninterrupted but arhgef2-C55R expression results in reduced AP length. Scale bar = 1 mm. (C) Reduced AP length and increased rates of pericardial edemas (D) appear in arhgef2-C55R expressing embryos (N = 23 control, 35 arhgef2-C55R). (E) Microaspiration-based compliance measurements show arhgef-C55R results in reduced tissue compliance (N = 8-9 over 2 clutches). (F) Upper panels shows immunofluorescence staining overlay of cardiomyocyte marker tropomyosin (red), epithelial marker aPKC (green) and rhodamine dextran (cyan). Lower panels show tropomyosin-masked aPKC expression in psuedocolor LUT, for emphasis. Scale bar is 50 µm. Quantification of apical aPKC expression shows arhgef2-C55R results in increased aPKC expression, while Y27632 treatment can reverse those effects (N = 7 to 11 embryos from each of 3 clutches). (G) Representative lateral view of immunofluorescence staining of stage 39 tadpole hearts; cyan = rhodamine dextran, red = troponin T, gray = fibrillin. Scale bar = 100 µm. Error bars represent mean ± SEM.

To further confirm the role of endoderm mechanics in inducing HPC MET, we targeted injections of a constitutively active form of the myosin binding subunit (MBS, MYPT1, or formally PPP1R12A) which we hypothesized would soften tissues. This constitutively active form, MBS-T695A, cannot be phosphorylated at one of two inhibitory sites, and has been reported to result in a two-fold increase in myosin phosphatase activity compared to the wild-type (Muranyi, Derkach et al. 2005). When MBS-T695A was expressed in the anterior endoderm, we noted whole embryo defects (Fig. 16), including reduced AP length (Fig. S4B) and increased rates of cardiac edemas (Fig. 16). Microaspiration of the HFR revealed compliance increased in MBS-T695A embryos by approximately 30% (Fig. 16). Similar to what we observed in blebbistatin and Y27632 treated embryos, we found aPKC intensity decreased by approximately 40% with the increased compliance in the endoderm (Fig. 16E and F). Calyculin A, which inhibits myosin phosphatase activity downstream of MBST695A, was able to rescue many of these defects, including the reduced AP length (Fig. 16B) and localization of aPKC (Fig. 16E and F).

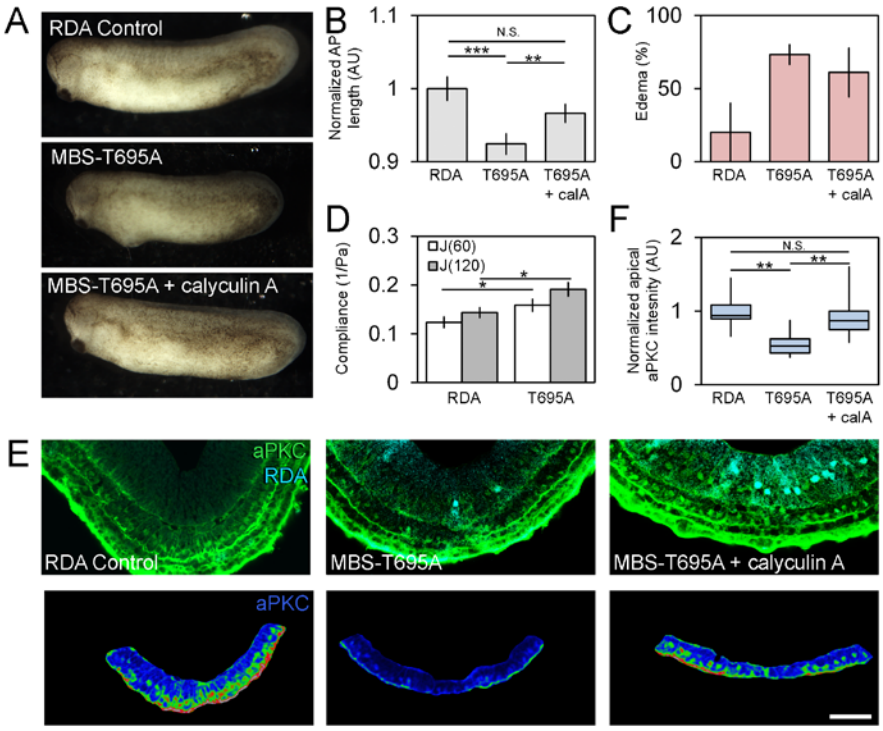

Figure 16. Targeted injections to perturb endoderm contractility with MBS-T695A increases compliance and reduces aPKC localization.

(A) Tadpoles injected with either rhodamine dextran (RDA) or a constitutively active MBS-T695A. (B) MBS-T695A injected embryos exhibit reduced anterioposterior length (N= 14-19 embryos over two clutches) and (C) have higher rates of edema presentation per clutch (N = 2 clutches). These defects can be partially rescues by treatment with calyculin A during Stages 20-28. (D) Embryos expressing MBS-T695A in their anterior endoderm have increased compliance in the HFR (N = 13-14 embryos over two clutches). (E) Top panels show immunofluorescent staining of Stage 28 embryos with aPKC (green) and fixed injected RDA (cyan). Bottom panel shows manually segmented HPCs with aPKC in pseudocolor LUT. (F) Normalized apical intensity measurements of aPKC shows reduced intensity in HPCs with MBS-T695A injected into endoderm, which can be rescued with calyculin A treatment (N = 6-8 embryos over two clutches). Scale bar = 50 μm. Error bars represent mean ± SEM. * denotes $p < 0.05$; ** denotes $p < 0.01$; *** denotes $p < 0.001$.

Next we tested whether the cell autonomous mechanics of HPCs might also play a role in regulating MET. Arhgef2-C55R targeted to HPCs displayed stronger aPKC localization than neighboring non-expressing HPCs (Fig. 17A, arrow). We found a significant correlation between levels of arhgef2-C55R expression, evaluated by intensity of the co-injected rhodamine dextran (RDA), and apical localization of aPKC (Fig. 17A) suggesting a graded increase in apicobasal polarity with increasing levels of autonomous contractility. Furthermore, as with the case of increased contractility in endoderm, HPCs expressing arhgef2-C55R treated with Y27632 showed little evidence of MET (Fig. 17A). By stage 39, tadpoles with arhgef2-C55R expressing cardiomyocytes displayed abnormal cardiac structure, with defects appearing centered around cells expressing arhgef2-C55R (Fig. 17B). Thus, mechanical perturbations targeted to either endoderm or HPCs confirm results obtained with small molecule inhibitors and identify a direct role for cell contractility and mechanics in regulating early MET.

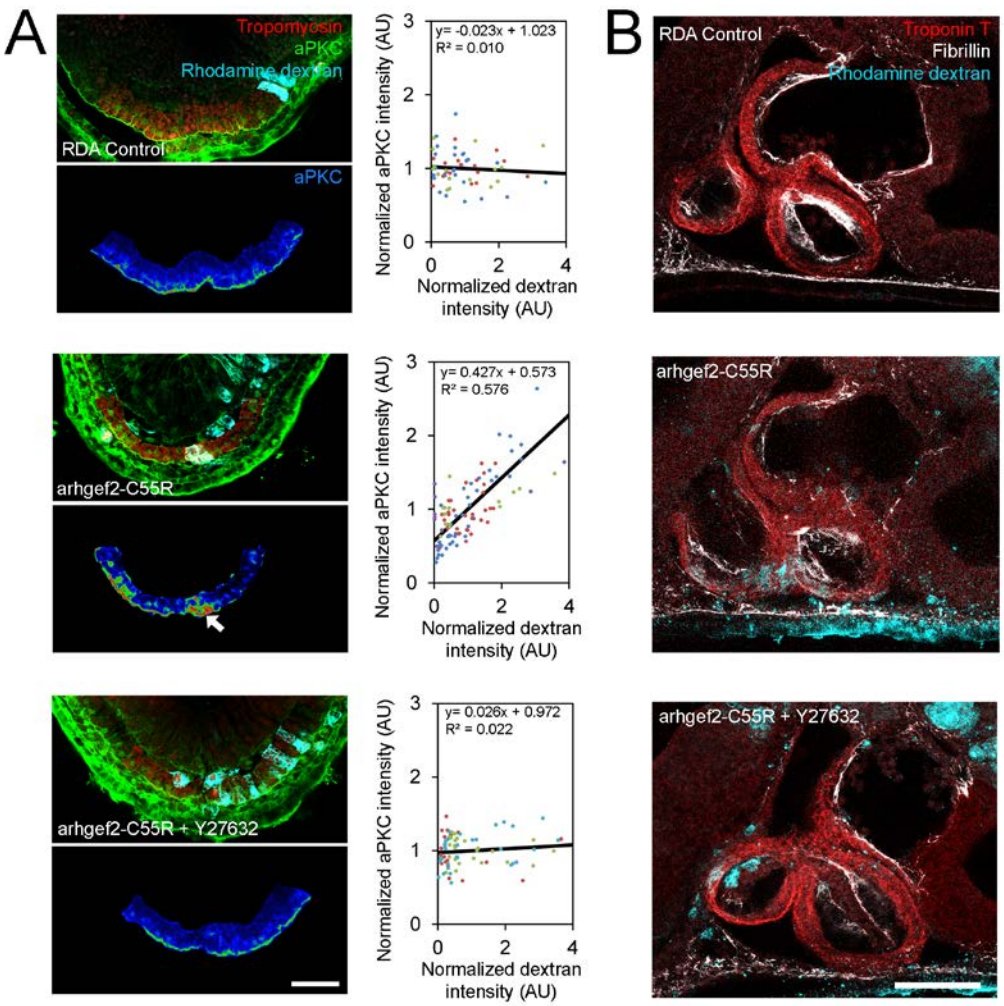

Figure 17. Targeted injections to perturb HPC contractility shows cell-autonomous mechanics influence MET.

(A) Transverse sections of stage 28 embryos stained for tropomyosin (red) and ZO-1 (green) with rhodamine dextran (RDA; cyan) showing expression of construct target injected into HPCs. Expression of arhgef2-C55R results in visibly elevated expression of aPKC (arrow). Quantification of individual cell apical intensities of RDA and aPKC show significantly correlation in arhgef2-C55R expressing HPCs but not RDA controls or embryos expressing arhgef2-C55R and treated with 50μM Y27632. Scale bar is 50 μm. (B) Stage 39 embryos with HPC-targeted expression of injected constructs. arhgef2-C55R embryos show relatively normal hearts with defects localized to where arhgef2-C55R is expressed. Treatment with Y27632 partially rescues these defects. Scale bar = 100 μm.

3.2.6 External mechanical tension can induce MET

Since cell contractility is a key contributor to junction formation between epithelial cells (Yamada and Nelson 2007, Lechuga, Baranwal et al. 2015, Bai, Zhu et al. 2016, Jodoin and Martin 2016), we wanted to test whether MET in HPCs could be driven purely by exogenous mechanical stress. To increase stress within the heart forming region we injected a 70 nl droplet of biologically inert mineral oil (~500 µm diameter) into the anterior cavity of the archenteron of just after neural tube formation (stage 21; Movie S4). Upon injection, the droplet produces a 5% stretch in the ventral ectoderm surface area (Fig 18A) and remains nearly spherical (Fig. 19A). Based on compliance measurements of the HFR at this stage (Fig. 14E), we estimate 0.3 Pa of stress is applied to the HFR. Droplet-injected embryos continue to elongate normally but develop cardiac edemas with high frequency (~80%; Fig 18B); embryos wounded with the microinjection needle (Sham control; Fig. 18B), or injected with a smaller 12 nl droplet (140 µm diameter) that produces no deformation, exhibit much lower rates of cardiac edema (Fig. 19B-C). MET increases within HPCs in large droplet-injected embryos by stage 28 (Fig 18C) with a 50% increase in apically localized aPKC and a nearly two-fold increase in ZO-1 similar to the case of embryos with stiffer endoderm (Fig 15F). Thus, we were able to show that physically-induced mechanical stress is sufficient to accelerate MET in HPCs without directly manipulating molecular signaling or actomyosin contractility.

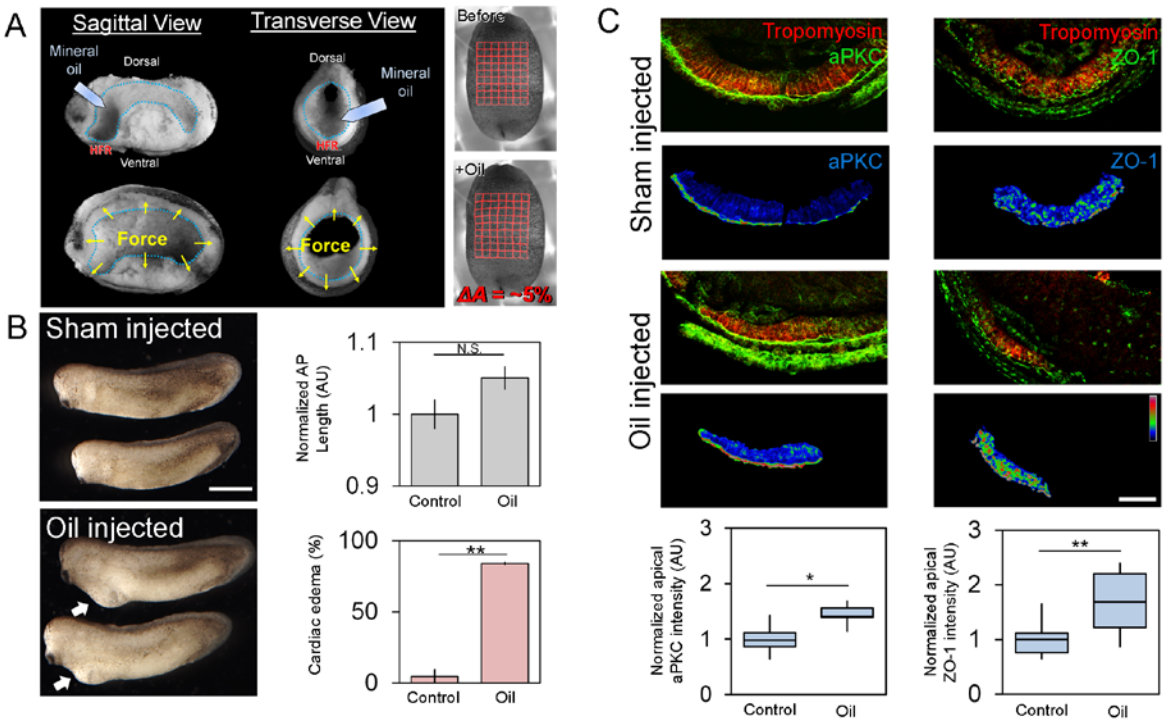

Figure 18. Exogenously applied stress can accelerate MET in HPCs.

(A) Schematic showing injection of biologically inert mineral oil into the foregut (outlined in blue). Sagittal and transverse views of embryos pre-injection (upper) and post-injection (lower) with forces indicated by yellow arrows. Right panel shows embryo pre-injection and post-injection. Ventral deformations tracked automatically show area stretch of approximately 5%. (B) Stage 28 embryos injected with oil show no defects in AP length (N = 13-14 embryos over two clutches) but a high frequency of ventral edemas per clutch (N = 2 clutches). Scale bar = 1 mm. (C) Immunofluorescence staining of transverse sections of the ventral midline. Upper panels show overlay of cardiomyocyte marker tropomyosin (red) and epithelial markers aPKC or ZO-1 (green) and lower panels show tropomyosin-masked epithelial marker expression with pseudocolor LUT for emphasis. Scale bar = 50 µm. Apical aPKC and ZO-1 localization increases after oil injection (N = 11 to 13 embryos over three clutches). Error bars represent mean ± SEM. * indicates $p < 0.05$ and ** indicates $p < 0.01$.

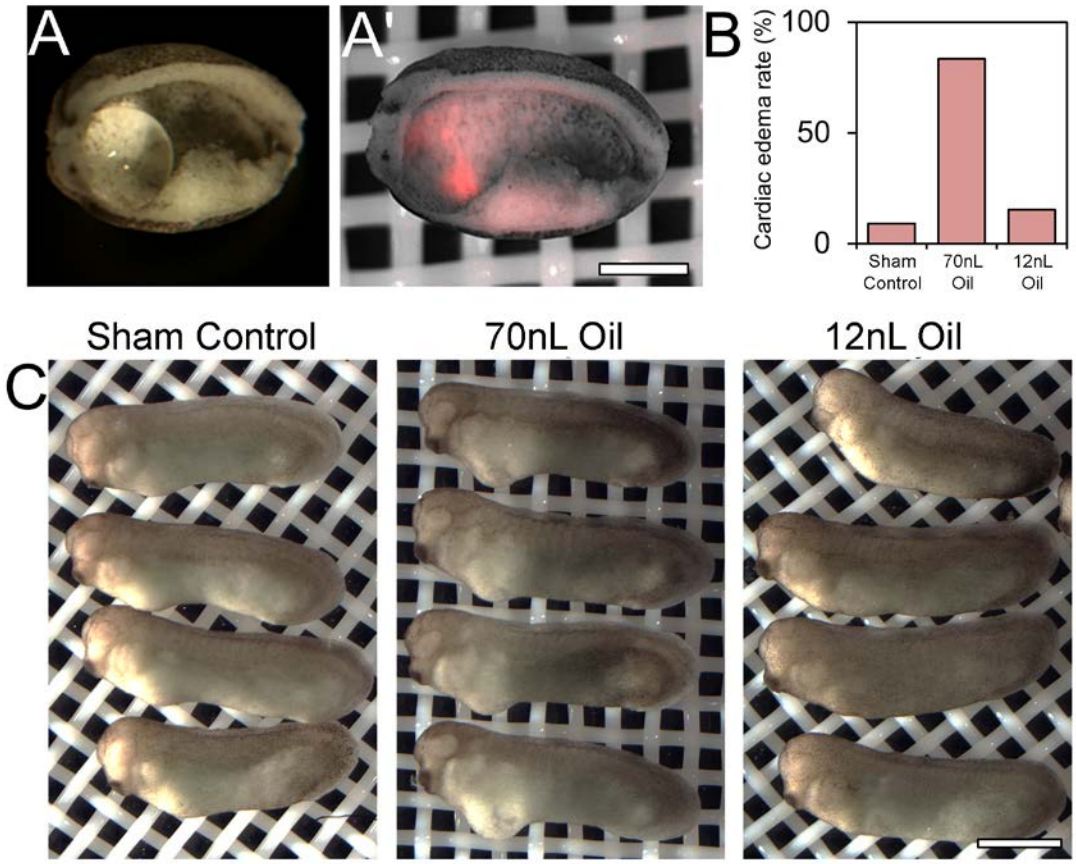

Figure 19. Cardiac defects resulting from mineral oil injects are caused by stress on the anterior ventral tissues.

(A) Embryo fixed and sectioned one hour after injection with mineral oil (~70nL) shows oil remains localized to the anterior portion of the embryo. (A') DiI dye transfer from mineral oil to cells shows anterior ventral endoderm tissues are in direct contact with the stress-inducing oil. Scale bar = 500 µm. (B) Edema rates show elevated cardiac defects in full amount mineral oil (~70nL) injected embryos but not sham controls or reduced amount oil (~12nL). (C) Embryos were either wounded with the microinjection needle (sham control), injected with the full amount of mineral oil (~70nL) or injected with one-sixth of the full amount of mineral oil (~12 nL), which did not cause expansion of ventral tissues. Scale bar = 1 mm.

3.2.7 Perturbing MET has physiological consequences in the larval beating heart

We were surprised by the specific and robust defects in cardiac architecture when MET was transiently disrupted and wondered if perturbations in the timing of MET translated to physiological defects in cardiac function. To assess cardiac function we visualized heart beat once the larval heart begins to drive circulation (stage 42) after a range of MET perturbations including: chronic modulation of contractility (stages 20 to 28, ~11 hours), endoderm-targeted injection of arhgef2-C55R, and stress-inducing mineral oil injections. Using hemoglobin contrast subtraction angiography (HCSA; (Deniz, Jonas et al. 2012)), which is analogous to an echocardiograph, we visualized blood flow through the ventricle (Movie S6; Fig. 7A; arrows indicate individual trabeculae). The area of hemoglobin blush was tracked over time (Fig 20A; see methods) allowing us to measure parameters of cardiac function including the ejection fraction (EF), heart rate (HR), end-diastolic area (A_{ed}), end-systolic area (A_{es}), and ejected area (A_{ej}; Fig. 20C and Fig. 21); control embryos have a mean EF of 91%, comparable to previous reports (Deniz, Jonas et al. 2012). Blebbistatin produced the most severe physiological defects with stationary blood islands superficial to the primitive gut (data not shown); while these embryos exhibited a regular heartbeat they had no blood flow. Both calyculin A and arhgef2-C55R yielded significantly larger A_{es} relative to the control, indicating that ventricular contraction was unable to efficiently pump blood out of the heart. Oil injected embryos presented highly variable heart morphologies, ranging from fairly normal to fully bifid hearts, with highly variable function. Y27632 embryos had mild defects with lower EF, lower HR and higher A_{es} but none were found to be statistically significant. Thus, perturbations in MET in the heart progenitor cells produce highly specific defects in cardiac architecture and distinctive defects in cardiac function.

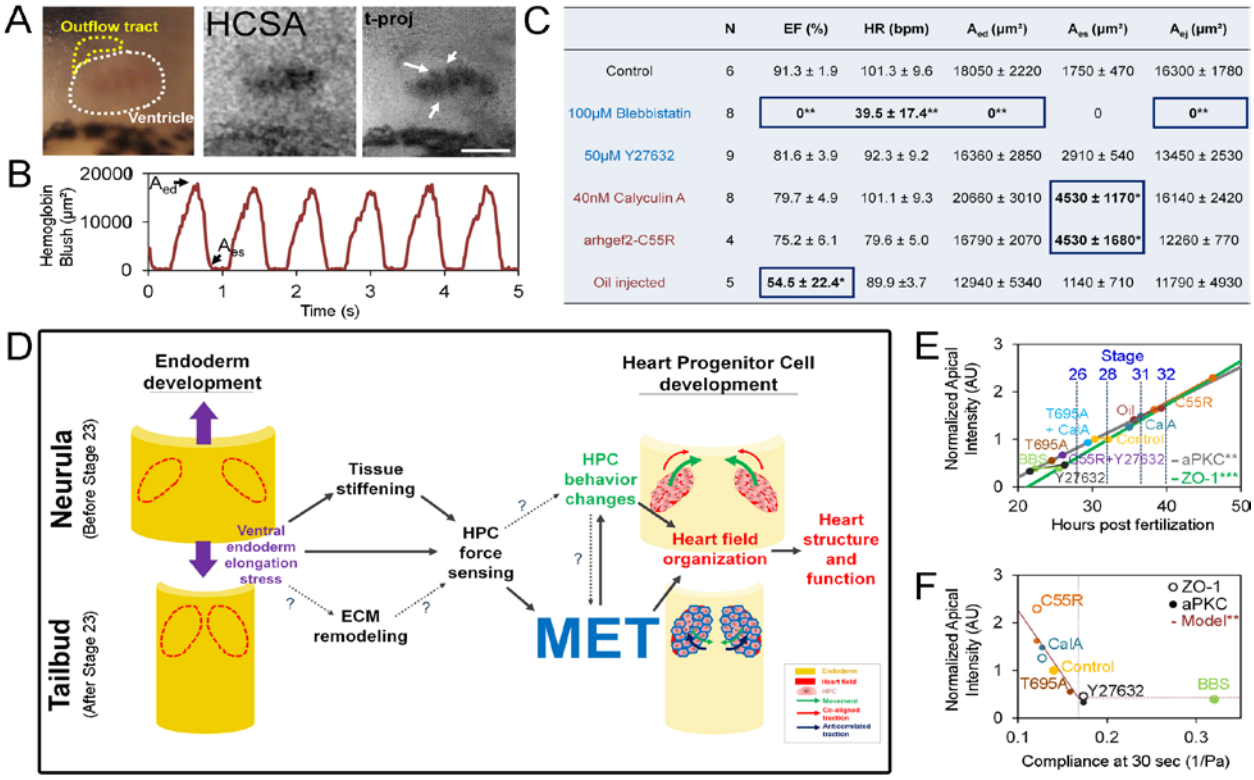

Figure 20. Larval heart function as a result of various mechanisms for MET perturbations.

(A) RGB image of 3-day old tadpole heart with ventricle outlined in white and outflow tract outlined in yellow. Hemoglobin contrast subtraction angiography (HCSA) image and multi-frame, time-projection of HCSA over one cardiac cycle with arrows indicating individual trabeculae being detectable. Scale bar = 100 μm. (B) Plot of hemoglobin blush overtime reveals typical cardiac cycle dynamics, with end-diastolic volume (Aed) and end-systolic volume (Aes) indicated on graph. (C) Table of parameters calculated. Data pooled from 3 clutches. EF = ejection fraction, HR = heart rate, Aed = end-diastolic area, Aes = end-systolic area, Aej = area ejected. * indicates $p < 0.05$; ** indicates $p < 0.01$. Blue font indicates treatments to increase compliance/decrease cell contractility; red font indicates treatments to decrease compliance/increase cell contractility and stress. (D) Schematic of the mechanical linkages between endoderm development, heart progenitor cell development, MET, and cardiogenesis. (E) Plot of endogenous epithelial marker intensity increase over time from Fig. 1F-I with various stage 28 mechanical perturbation data from Figs. 2, 5, 6 fit onto regression line (aPKC: y = 0.77t - 1.34; R2 = .446; ZO-1: y = 0.92t - 1.96; R2 = .567). (F) Plot of compliance versus normalized apical intensity with regression fit to data from arhgef2-C55R, calyculinA, control and Y27632 data (y = -27.22*J(60) + 4.99; R2 =.843).

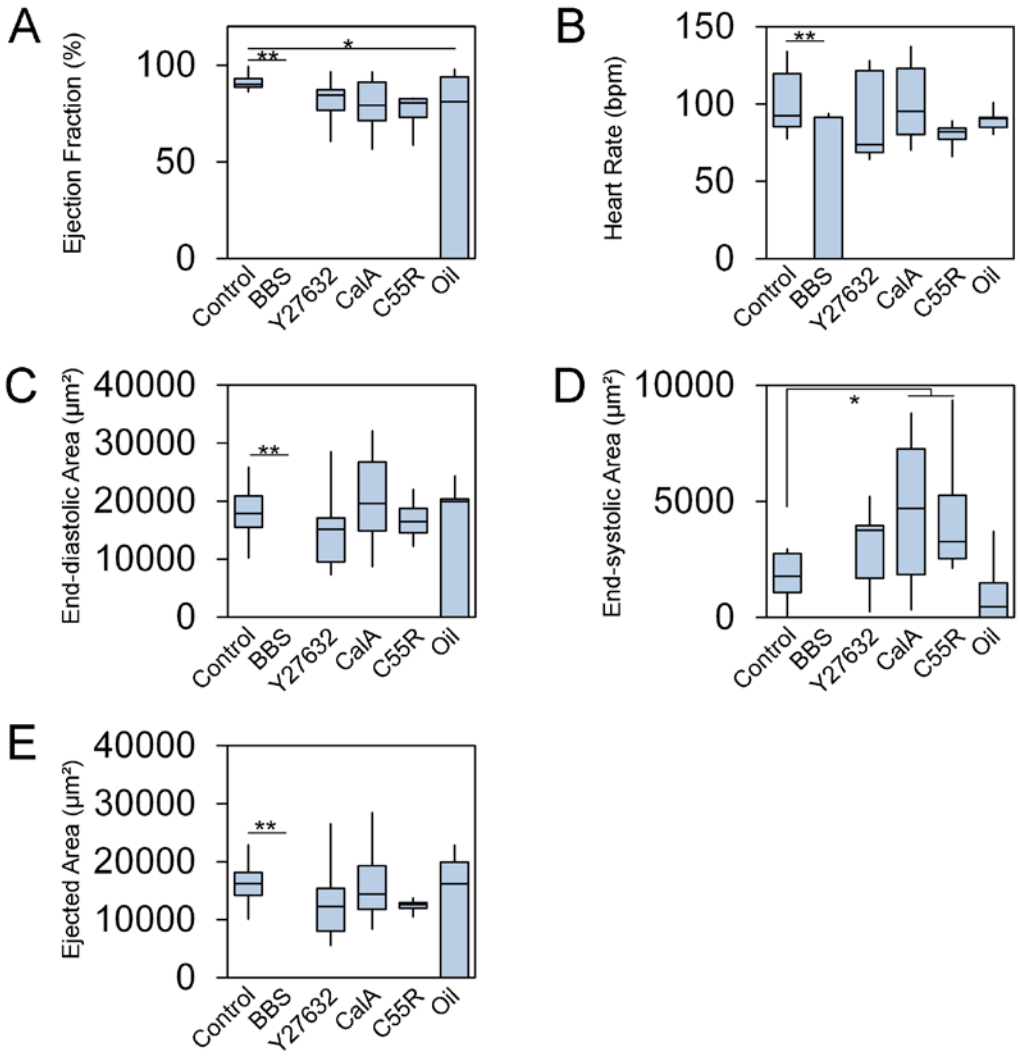

Figure 21. Graph representations of hemoglobin contrast subtraction angiography data.

Boxplots shown for (A) ejection fraction, (B) heart rate, (C) end-diastolic volume, (D) end-systolic volume and (E) ejected volume. * denotes $p < 0.05$; ** denotes $p < 0.01$.

3.3 DISCUSSION

On their way to the ventral midline HPCs traverse a dynamically changing mechanical microenvironment marked by fibronectin remodeling (Yelon and Stainier 2005, Garavito-Aguilar, Riley et al. 2010, Ye, Xie et al. 2015), changing tissue stiffness (Zhou, Kim et al. 2009, Shi, Varner et al. 2015), and tension from ventral convergent extension (Larkin and Danilchik 1999, Varner and Taber 2012, Shi, Varner et al. 2015). In frog we propose that HPCs utilize mechanosensors to integrate these mechanical cues on their way to the ventral midline and respond by initiating MET (Fig. 20D). Mechanical cues sensed by HPCs reflect changes in endoderm compliance, either as developmentally programmed or in some form of gradient, and may also reflect changes in the mechanical properties of other tissues composing the HFR. In order for HPCs to sense these cues they must be able to actively probe their microenvironment, using contractility to actuate their internal mechanosensing pathways. While MET is not prerequisite for HPCs to reach the ventral midline, the proper timing of MET is required for normal heart structure and function (Fig. 20C and Fig. 21). We support this proposal with evidence collected from global-, HFR-specific, endoderm-specific, and HPC-specific mechanical perturbations. By modulating bulk tissue compliance in the HFR (Fig. 10) or specifically in the endoderm (Fig. 15) we show that HPCs can sense and respond to mechanical cues to either block or accelerate MET. Mechanical cues must be under tight temporal control since the window for proper MET is limited and sensitive to perturbation (Fig. 10I and 11). Perturbations that increase compliance mimic conditions of early stages whereas treatments that decrease compliance or induce mechanical stress mimic mechanical conditions found at later stages (Fig. 20E). From both description of developmental changes and experimental perturbation studies we find HFR tissue compliance and apicobasal polarity display an inverse relationship (Fig. 20F). An

exception to this trend is seen at very high levels of compliance where HPCs fail to polarize (Fig. 20F; purple line). Once the bilateral fields of HPCs assemble at the midline mechanics plays a more conventional role, folding the tube and establishing stereotypic left-right asymmetry (Zamir and Taber 2004, Shi, Varner et al. 2015).

Heart defects are common following perturbations of actomyosin during early heart development with the lesions commonly attributed to defects in cell migration (Wei, Roberts et al. 2001). Prior to MET, HPCs appear to be carried, or advected, ventrally by endodermal movements (Fig. 13). After MET, traction force patterns generated by moving HPCs suggest they migrate more autonomously into the ventral heart forming region (Fig. 13F-H). While it is often assumed that mesenchymal and epithelial cells exhibit distinctive migratory behaviors (Friedl and Gilmour 2009), recent studies have described a number of common behaviors. For instance, mesenchymal cells can undergo collective migration during development (Szabo and Mayor 2016, Szabo, Melchionda et al. 2016) in a manner analogous to epithelial cells (Ewald, Brenot et al. 2008, Hamm, Kirchmaier et al. 2016). While it is not clear whether epithelial or mesenchymal cancer cells migrate faster (Schaeffer, Somarelli et al. 2014), mouse breast cancer cells show increased motility *in vitro* following MET (Dykxhoorn, Wu et al. 2009). By contrast, we find the opposite behavior in HPCs *in vivo* (Fig. 13B-E). A consensus view is emerging that a phenotypic change, such as an EMT or MET, will have consequences for migratory behaviors but the exact consequences may vary across cell types and their microenvironmental context.

The epithelium formed by heart progenitor cells en route to the ventral midline is unique in anatomy and its contribution to the larval heart and suggests that MET occurs at a critical moment as bilateral fields of HPCs assemble into a single medial plate. The pre-cardiac mesoderm moves as a bilayer, with the apical surface of the epithelium appearing in the middle

of the two layers (Fig. 8B). Each of these two layers has a distinct fate with the deep layer giving rise to endocardial cells and cardiomyocytes (Linask 1992, Linask and Lash 1993) and the superficial layer likely contributing to the pericardium (von Gise and Pu 2012). The initiation of apical junctions between two layers of mesoderm is a likely first step in pericardial coelom formation (Linask 1992) as separation of inner and outer mesoderm layers splits precardiac mesoderm from ventral body wall mesoderm (Linask, Knudsen et al. 1997). Furthermore, because this epithelialization is transient and mature cardiomyocytes do not form an epithelium (von Gise and Pu 2012), the function and timing of MET likely plays a key role in establishing the structural foundation of the vertebrate heart.

Establishing a causal link between transcriptional regulation of HPC MET and how those changes alter HPC behaviors will require a detailed mechanistic analysis. HPCs and endoderm develop in such close association that endodermal movements carry HPC to the ventral midline (Aleksandrova, Czirok et al. 2015, Xie, Ye et al. 2016). Furthermore, communication between the endoderm and heart progenitors is critical for cardiac differentiation (Nascone and Mercola 1995, Lough and Sugi 2000). Several signaling networks have been implicated in the polarity establishment of HPCs, including PRKCI and Crumbs (Rohr, Bit-Avragim et al. 2006), hand2 and fibronectin (Garavito-Aguilar, Riley et al. 2010) and Slit and Robo (Medioni, Astier et al. 2008). Much is known about HPC movements (Zamir, Czirók et al. 2006), including their multiphasic nature (Ye, Xie et al. 2015, Xie, Ye et al. 2016) and coordination with endoderm (Aleksandrova, Czirók et al. 2012, Varner and Taber 2012, Aleksandrova, Czirok et al. 2015), however, little is known about role of MET. Interestingly, neither high compliance nor progression through a MET are required for HPC differentiation or their bulk movement to the ventral HFR (Fig. 10F) even though decreasing compliance and accelerating MET did

occasionally result in bifid hearts (Fig. 15F; Fig. 18C). Interestingly, convergent extension of the endoderm during chicken cardiogenesis creates a mediolateral stiffness gradient (Shi, Varner et al. 2015), which suggests HPCs experience higher stiffness or tension as they approach the ventral midline. Such stiffness gradients can generate cytoskeletal polarity and directional migration in mesenchymal cell culture (Raab, Swift et al. 2012, Trichet, Le Digabel et al. 2012) while similar mechanical patterning systems may contribute to cell polarity (Chien, Keller et al. 2015). It was recently discovered that *Xenopus* neuronal axons use stiffness gradients as cues for pathfinding (Koser, Thompson et al. 2016) and it is tempting to speculate that a similar set of cues might contribute both to the ventral movements of HPCs and the temporal control of MET. Thus, the biomechanical processes that shape the ventral endoderm are not only required to position HPCs, but also establish the apicobasal polarity and epithelial identity in these cells.

To explore the universal role of mechanotransduction during MET will require comparative analyses of the mechanical conditions that prevail during METs across different species, tissues, and stages. The absolute mechanical properties of embryonic tissues appear specific for each organism, tissue and age. For instance, the mechanical properties of gastrulating *Xenopus* and sea urchin embryos can differ by a 1000-fold (Davidson, Oster et al. 1999), avian HPCs at the early tube stage are stiffer (Shi, Varner et al. 2015), and later stages of the developing mouse heart are stiffer still (Majkut, Dingal et al. 2014). Still, the developmental profiles of different species share common trends (Zamir and Taber 2004); embryos stiffen as they age (Zhou, Kim et al. 2009) and mechanical properties can vary as much as two-fold between siblings (von Dassow and Davidson 2009). Elucidating the role of mechanical cues during MET will require broader application of biomechanical methods to more diverse model systems.

At present, we do not know how HPCs sense changes in mechanics although there are a number of likely candidates (Miller and Davidson 2013). Cilia play mechanosensory roles in kidney and vasculature development (Oh and Katsanis 2012, Battle, Ott et al. 2015) and a variety of ciliopathies contribute to cardiac defects (Kennedy, Omran et al. 2007, Duncan and Khokha 2016). Mechanosensing by filopodia appear to play a key role in outflow tract morphogenesis (Francou, Saint-Michel et al. 2014). Yet another possibility is that mechanical cues are sensed by focal adhesions to fibronectin mediated by α5β1 integrin since both required for movements of the HPC and endoderm (Trinh and Stainier 2004, Garavito-Aguilar, Riley et al. 2010) and appear to polarize cardiomyocytes along the mesoderm-endoderm interface (Linask 2003). Fibronectin null-mutants in Zebrafish disrupt both mechanical signaling and migration (Trinh and Stainier 2004); however, when force transduction is decoupled from signaling, pericardial edemas form that are similar to those found after MET is disrupted (Rozario, Dzamba et al. 2009). Future mechanistic studies are warranted in order to test the role of putative mechanosensors used by HPCs in the activation of MET.

Early mechanical events during development can play a critical role in shaping organs and enabling their physiological function. In the case of HPC MET this role is highly evident when a brief treatment with blebbistatin, transiently disrupting myosin II contractility, disorganized heart fields prior to merging, disrupted looping and the integration of the heart to the systemic circulation. HCSA analysis, analogous to an echocardiograph, revealed that cardiac function in embryos undergoing precocious or accelerated MET is characterized by a significantly larger end-systolic area without a change in ejection fraction (Fig. 20C and 21), which is a common feature of congenital heart defects (Abualsaud, Lowe et al. 2016) including mitral or aortic valve insufficiency (Sutton, Plappert et al. 1987, Hundley, Li et al. 1995) or

defects to the ventricular wall strength, including hypertrophy (Song, Dyer et al. 2011). Since precocious or accelerated MET in heart progenitor cells produces a distinct phenotype, future studies may uncover the relationship between genetic mutations known to cause congenital heart defects and causative defects in mechanical signaling in early heart development.

3.4 EXPERIMENTAL PROCEDURES

3.4.1 Xenopus embryos, microsurgery and microinjections

Xenopus laevis embryos were obtained via standard methods (Kay and Peng 1991) and cultured in 1/3× modified Barth Solution (MBS; (Sive, Grainger et al. 2000))). Both intact embryos and tissue explants can be used to expose development of heart progenitors for high resolution imaging (Fig. 22A). *Ex vivo* tissue isolates consisting of the ventral mesoderm and the bilateral fields of anterior lateral plate mesoderm are microsurgically removed from the embryo (Fig. 22B) and cultured in Danilchik's For Amy (DFA; (Sater, Steinhardt et al. 1993) media supplemented with antibiotic and antimycotic (Sigma). Amphibian *ex vivo* tissue isolates have a long history in exposing fundamental principles of developmental biology. Amphibian tissue isolates have been and continue to be a cornerstone of reductionist methods used to expose critical signaling pathways and routes of cell-cell communication. Without these methods it would have been extremely challenging to isolate specific pathways and principles of signal

transduction, cell adhesion, and morphogenesis. Tissue isolates are useful in biomechanical studies for the same reason they are useful for biochemical studies. Firstly, tissue isolates are composed of thousands of cells that retain key cell-cell communication pathways, cell-cell and cell-ECM adhesions, and large scale microanatomy found in the intact embryo. We have confirmed that post-gastrulation stage tissue isolates are fully capable of engaging programs of morphogenesis and generating structures that are indistinguishable from those structures produced in vivo and we expect both biochemical and biomechanical processes continue to operate as they would in vivo. Our specific application of tissue isolates of the heart forming region is based on classical applications of microsurgical methods to identify patterning factors operating in gastrula stage embryos (Nascone and Mercola 1995, Schneider and Mercola 1999). Our assumption that mechanical conditions in isolates approximates those in vivo are based on our work on the material properties and forces generated within tissue isolates over a series of biomechanical studies we have conducted both in vivo (von Dassow and Davidson 2009, von Dassow, Strother et al. 2010, von Dassow, Miller et al. 2014, Feroze, Shawky et al. 2015) and with ex vivo tissues (Moore 1994, Moore, Keller et al. 1995, Zhou, Kim et al. 2009, Zhou, Kim et al. 2010, Zhou, Pal et al. 2015). Here, our tissue isolates form a differentiated heart expressing troponin T and fibrillin and having similar anatomy to the *in vivo* heart (Fig. 22E).

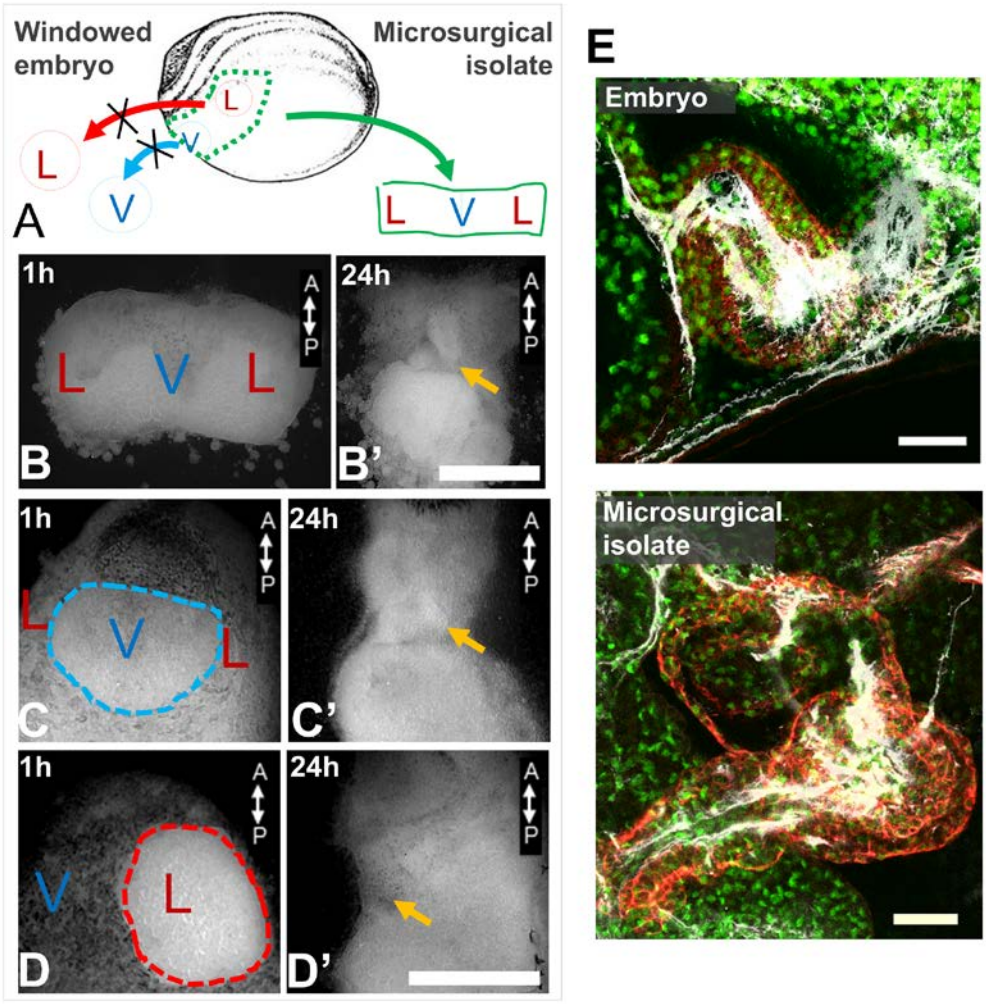

Figure 22. Microsurgical manipulations to visualize HPCs in *Xenopus.*

(A) Schematic of microsurgical manipulations to visualize HPC migration. (B) Microsurgical isolates at 1 h and 24 h. (C) Ventral windowed embryo at 1 h and 24 h. (D) Lateral windowed embryo at 1 h and 24 h. Yellow arrows mark location of beating heart. Scale bar = 500 μm. (E) Stage 33 embryo and equivalent stage microsurgical isolate stained for troponin T (red), fibrillin (white) and nuclei (green). Scale bar = 50 μm.

In addition to *ex vivo* isolates, intravital "windowed" embryos can be generated by microsurgically removing the ventral ectoderm (Fig. 22C). mRNA encoding fluorescent protein constructs were injected at the one- or two-cell stage to visualize nuclei (H2B-mCherry), plasma membrane (CAAX-GFP) or F-actin (moesin-GFP) . Targeted injections of mRNA to endoderm or HPCs were carried out at the 32-cell stage based on established *Xenopus* fate mapping protocols (Dale and Slack 1987, Bauer, Huang et al. 1994). mRNA encoding arhgef2-C55R or MBS-T695A was co-injected with rhodamine dextran amine (RDA) for visualization purposes. Accuracy of targeted injections was verified visually by RDA prior to fixing embryos at Stage 28.

3.4.2 Small-molecule inhibitor treatments

In order to manipulate tissue stiffness through cell contractility, embryos were either subjected to a chronic (from Stage 20 to 28) or acute (Fig. 11) treatment with Y27632 (50 µM), Blebbistatin (100 µM), or Calyculin A (50 nM).

3.4.3 Histology, immunostaining and confocal microscopy

Tissue isolates or whole embryos were fixed with either 4% paraformaldehyde for 2 hours or Dents fixative overnight. Posterior halves of embryos were removed prior to staining to minimize antibodies getting stuck in gut cavity. After blocking, immunofluorescence staining was carried out with primary antibodies JB3, CT3, CH1 (Developmental Studies Hybridoma Bank), aPKC (Santa Cruz), ZO-1 (Invitrogen), 4H2 (courtesy of Douglas DeSimone), β-catenin (Sigma-Aldrich). Appropriate secondaries (1:1000; Jackson ImmunoResearch) were then used to

stain overnight. Nuclei were identified with YO-PRO-1 (ThermoFisher). Transverse sections of embryos were generated either using a vibratome or manually using a scalpel and then dehydrated and cleared. Confocal stacks were collected using a laser-scanhead mounted on an inverted compound microscope (Leica SP5). For quantification purposes, acquisition parameters were kept constant throughout experiments and slices in stacks were collected at 1 μm z-intervals. To quantify apical intensity of epithelial markers, stacks were masked using the tropomyosin channel to ensure only cardiomyocytes were considered, normalized to the background fluorescence in the endoderm cells, summed over consecutive 5 μm z-volumes and then blurred using a small kernel Gaussian blur (Fig. 9). Intensity along a 10-pixel thick linear region (Fig. 9) along the apical surface was measured and measurements were done at 4 z-depths for each embryo (Fig. 9). For the acute small-molecule inhibitor treatment studies (Fig. 10I and Fig. 11), fluorescence images were acquired with CCD camera (QImaging) mounted on a fluorescence-equipped stereomicroscope (Olympus). Images were then double-blinded for ROI drawing around the tadpole heart. The cardiac morphology index was calculated in order to consider both heart size and shape, using the area and circularity obtained from the drawn ROI (Fig. 11).

3.4.4 Compliance measurements using micro-aspiration

To measure the mechanical properties of the heart forming region, embryos were placed in the high-pressure reservoir of a dual-reservoir micro-aspirator apparatus (Fig. 10D, (von Dassow and Davidson 2009, von Dassow, Strother et al. 2010, Kim, Jackson et al. submitted)). Embryos were gently compressed using hair tools such that the ventral heart forming region (just

medial of the ventral midline) was up against a 125 μm diameter channel cast across a PDMS block to form a seal across the opening of the channel. A pressure differential at the channel opening was controlled hydrostatically using a computer-controlled syringe pump. After calibrating the channel to obtain zero flow, a small baseline pressure of approximately -1.5 Pa was applied in order to initiate suction and seal the embryo onto the channel opening. To permit tissue relaxation prior to testing, we allowed a 5 minute pause before the 155-second micro-aspiration protocol was initiated. At the start of the protocol, the embryo was imaged at 1 frame per second to obtain an initial aspiration length for the tissue. After 5 frames, the syringe pump removed 800 μL of media from the low-pressure reservoir resulting in a final -10.1 Pa suction pressure applied to the aggregate. After 120 seconds, 800 μL was media was replaced to return to the baseline pressure. Imaging continued for an additional 30 seconds to confirm the embryo was properly sealed onto the channel opening. If carrying out repeated measurements on the same sample, the embryo was rotated to a different position. The tissue boundary in the microchannel was tracked either manually or automatically with Canny-Deriche edge detection (Schneider, Rasband et al. 2012). The aspirated length was measured and the data was fit to the power-law model for creep compliance, which has previously been shown to provide an adequate fit for embryonic tissue responses to stress-application (von Dassow, Strother et al. 2010). Time-dependent compliance at 60 seconds (fast response) and 120 seconds (steady-state) is reported as the measurement of embryo mechanical properties. Compliance calculations for each sample are independent of the magnitude of the pressure applied (von Dassow, Strother et al. 2010). The suction force of -10.1 Pa was chosen in order to sufficiently displace aggregate tissues within the resolution of our imaging system without exceeding the critical pressure (P_c), thus ensuring measurements are representative of viscoelastic properties (Sato, Levesque et al. 1987). To

calculate the stress index in Fig. 14F, strain rate values were taken from the data in Fig. 14D at the developmental stages corresponding to the compliance measurements in Fig. 14E: 19 (2h), 21 (4h) and 23 (6h). Compliance values were inverted to obtain equivalent modulus and then multiplied by the strain rate to calculate a stress index.

3.4.5 Low-resolution traction force microscopy

Ex vivo ventral tissue isolates expressing a membrane-targeted GFP (CAAX-GFP) were cultured on a 0.05% polyacrylamide gel conjugated with human fibronectin (Sigma) containing red and far red beads (Molecular Probes) following an established protocols (Leach, Brown et al. 2007, Zhou, Kim et al. 2010). Tissue isolates were allowed to adhere to the gel and develop for 2 hours prior to imaging. Live imaging of cell movement and bead displacement was done on a laser-scanning confocal microscope (Leica SP5, Leica Microsystems). At the conclusion of the experiment, tissue isolates were exposed to Trypsin to induce cell release from the gel, in order to obtain a zero force image of the gel. Customized implementation of rigid registration algorithms from the Insight Toolkit (ITK; (Yoo, Ackerman et al. 2002)) was used to register the time-lapse sequences of the cell and each individual bead channel to the zero force image, using a portion of the gel without cells over top as a landmark for registration. Bead and cell displacements were calculated using a finite element registration algorithm from the Insight Toolkit, registering each consecutive frame to the zero force images. Displacements in the two separate bead channels were averaged together to improve accuracy (Sabass, Gardel et al. 2008). Bead displacements and cell displacements were averaged over 30 minute increments, in order to remove oscillations from individual cell protrusion and release cycles. Cosine similarity scores

were calculated using a custom macro written in ImageJ (Schneider, Rasband et al. 2012). See Section B.2 in the Appendix for more details.

3.4.6 Long-term time-lapse nuclei tracking

Nkx2.5-GFP transgenic embryos were injected with a Histone-2B-mCherry mRNA. Using the intravital ventral "windowed" embryo model, we image mesoderm development over 24 hours until the GFP signal was detectable to allow for positive identification of HPCs. Using the nkx2.5-GFP intensity to create a mask for the last frame of the time-lapse sequence, nuclei were tracked retroactively using a combination of watershed segmentation and finite element registration image filters from ITK. In brief, nuclei were segmented out and consecutive frames were registered to calculate predictive nuclei displacement values. Those predictive displacements were applied to a nucleus centroid and then nuclei in the next frame were matched based on minimum distance from that predictive value. Information on unmatched nuclei was saved to retry matching for up to the next 6 frames (1 hour). Persistence was calculated over thirty minutes as the effective distance traveled over the total distance. Collectiveness is defined as the cosine similarity between motion vectors of a given nuclei and all of the other nuclei within its "neighborhood", defined as within a 40 μm radius. Average values of parameters were calculated for each nucleus over the time periods defined in Fig 3H. Section B.1 in the Appendix for more details.

3.4.7 Mineral oil injections

Nkx2.5-GFP transgenic embryos were injected with a Histone-2B-mCherry mRNA. Using the intravital ventral “windowed” embryo model, we image mesoderm development over 24 hours until the GFP signal was detectable to allow for positive identification of HPCs. Using the nkx2.5-GFP intensity to create a mask for the last frame of the time-lapse sequence, nuclei were tracked retroactively using a combination of watershed segmentation and finite element registration image filters from ITK. In brief, nuclei were segmented out and consecutive frames were registered to calculate predictive nuclei displacement values. Those predictive displacements were applied to a nucleus centroid and then nuclei in the next frame were matched based on minimum distance from that predictive value. Information on unmatched nuclei was saved to retry matching for up to the next 6 frames (1 hour). Persistence was calculated over thirty minutes as the effective distance traveled over the total distance. Collectiveness is defined as the cosine similarity between motion vectors of a given nuclei and all of the other nuclei within its “neighborhood”, defined as within a 40 μm radius. Average values of parameters were calculated for each nucleus over the time periods defined in Fig 13C.

3.4.8 Assessment of cardiac function

To assess the cardiac function of a larval heart, we adapted a protocol previously established (Deniz, Jonas et al. 2012). Briefly, embryos were cultured three days and embedded in 3% ultra-low melting point agarose. Time-lapse sequences of heartbeats were recorded 60 frames-per-second using a DSLR camera (Canon T3i) mounted onto the video port of a fluorescence-equipped stereoscope (Olympus). A GFP filter was used to further enhance native

hemoglobin contrast. Angiography was accomplished by applying custom image analysis macros to the RGB time-lapse sequences (ImageJ; (Schneider, Rasband et al. 2012)). RGB videos were separated into red and green channels. Red channels were multiplied by a constant to achieve the same mean intensity as the green channel. The green channel was subtracted from the red channel producing a time-lapse sequence of hemoglobin contrast subtraction angiography images (HCSA) which were auto-thresholded to highlight relative changes in hemoglobin (hemoglobin blush). Quantitative time-series analysis of hemoglobin blush generated standard echocardiograph-like parameters of cardiac function (Deniz, Jonas et al. 2012) that were averaged over at least 15 cardiac cycles for each embryo.

3.4.9 Statistical analysis

For each experiment, an ANOVA was first done to determine if the clutch-to-clutch variation was significant to confirm that data from multiple clutches could be pooled. An ANOVA was then performed on pooled data, with a planned simple contrast to compare the appropriate control to each treatment. Correlations for traction co-alignment over time (Fig. 13D), dextran-aPKC colocalization (Fig. 17A) and normalized apical intensity over time and compliance (Fig. 20E-F) was estimated by fitness of a simple linear regression. Statistical analyses were performed using IBM SPSS Statistics 22. Graphs were produced with Microsoft Excel, with information about sample size and the error bars located in each figure legend.

4.0 3D TISSUE MODEL FOR MESENCHYMAL-TO-EPITHELIAL TRANSITION REVEALS INFLUENCE OF MECHANICS ON CELL PHENOTYPE”

The following chapter is a portion of a submitted manuscript. Here, I expand upon Specific Aims 2 to investigate how those mechanical cues regulate MET in mesenchymal cell aggregates. The original article was written in close collaboration with Hye Young Kim. The version here has been revised to emphasize my contributions to the manuscript.

The mesenchymal-to-epithelial transition (MET), an ability of loosely associated migratory cells to form a compact epithelium, is a conserved motif occurring throughout organogenesis and plays a key role in cancer metastasis and the generation of iPSCs. Here we introduce a three-dimensional (3D) tissue model to study mesenchymal-to-epithelial transition using Xenopus laevis embryonic mesenchymal cell aggregates and demonstrate a key role for tissue mechanics in initiating and regulating MET (Fig. 23). Within 5 hours post aggregation, mesenchymal cells along the aggregate surface form a cohesive epithelium complete with apicobasal polarity and epithelial markers including keratin and ZO-1, and complete their differentiation into epidermal goblet cells. Live imaging of GFP-ZO-1 reveals a stochastic transition of individual mesenchymal cells which assemble into multi-cell clusters that join to cover the entire aggregate surface. This 3D tissue platform enables quantitative assessments of tissue compliance, YAP nuclear localization, and MET progression that allow us to propose cell contractility and cell-cell adhesion modulate the physical properties required for MET.

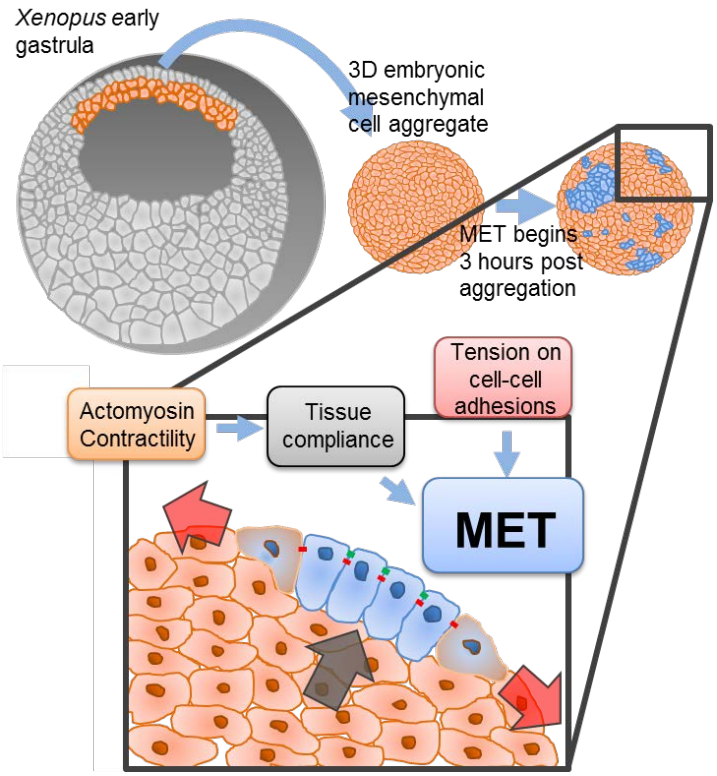

Figure 23. Schematic of our 3D cell aggregate model for studying mechanical regulation of MET.

4.1 INTRODUCTION

Cells in our body are highly plastic in both with regard to their phenotype and their mechanical properties. For instance, a tightly adherent epithelial cell in a sheet may undergo a transition into a loosely bound mesenchymal cell allowing it to migrate into surrounding tissues (e.g. an epithelial-to-mesenchymal transition or EMT). Conversely, motile mesenchymal cells can undergo a mesenchymal-to-epithelial transition (MET) during which they aggregate and form an organized epithelium. Such transitions involve large scale changes at the level of transcription, the cytoarchitecture, and cell behavior. MET appears to have evolved as a revolutionary strategy

in the metazoan lineage (Pérez-Pomares and Muñoz-Chápuli 2002) where it plays a fundamental role in many tissue-shaping processes including cancer (Sosa, Bragado et al. 2014), development (Stark, Vainio et al. 1994, Trinh and Stainier 2004) and stem cell reprogramming (Li, Liang et al. 2010, Samavarchi-Tehrani, Golipour et al. 2010). METs are key to organogenesis of the kidney (Dressler 2002) and the heart (Trinh and Stainier 2004, Trinh, Yelon et al. 2005) as clusters of mesenchymal cells migrate great distances in the embryo and transition to epithelial phenotypes to form the simple tubule of the developing kidney or a disk of the heart primordia. The timing and progression of MET during development parallels the events leading to the formation of secondary metastases (Chaffer, Thompson et al. 2007) and suggest these processes share common regulatory mechanisms (Nieto 2013).

Despite the significance of MET in topics ranging from cell biology to human health, little is known about the spatiotemporal dynamics of MET progression or how the physical regulation of MET integrates with conventional intracellular molecular signaling pathways (Kim, Jackson et al. 2016). The limited understanding of the cellular and molecular mechanisms of MET directly result from both the inherent unpredictability of MET emergence deep within multiple layers of cells *in vivo* and a lack of a tractable 3D model system that can recapitulate the full process of MET. By contrast, the past decade of research has provided great insight into the regulatory mechanisms of EMT due to established *in vitro* and animal models that permit profiling differential gene expression, tracing cellular transitions, and manipulating signaling factors (Yang and Weinberg 2008, Thiery, Acloque et al. 2009). MET is often viewed as the reverse of EMT; however the complicated sequential progression of MET cannot be reversed by reintroduction of mesenchymal inducing factors alone (SJ Bidarra et al 2016). These gaps in our understanding of MET indicate a clear demand for a model system that can be used to explore

the full progression by which mesenchymal cells initiate and progressively adapt apicobasal polarity, propagate “epithelialness” and stabilize the new architecture of epithelium (Kim, Jackson et al. 2016)

The physical mechanics of the microenvironment change prior to and concomitant with MET, as dispersed migratory mesenchymal cells first cluster then spontaneously develop cell-cell adhesions. As MET progresses, transitioning tissues often form compact spherical aggregates, disks, or cords. Mechanical tension, driven by cortical actin contraction and cell-cell adhesion, is known to shape cells and tissues during aggregation *in vivo* (Oster, Murray et al. 1983) and *in vitro* (Mammoto, Mammoto et al. 2011, Bhumiratana, Eton et al. 2014) and has been implicated in other aspects of tissue self-assembly (Steinberg and Gilbert 2004), cell rearrangement (Harris 1976, Maitre, Berthoumieux et al. 2012), and cell fate determination (Engler, Sen et al. 2006). To understand the role of tension in these processes, numerous methods for measuring tension sensed by cells have emerged (Shawky and Davidson 2015, Campàs 2016), including FRET-based tension sensors (Cost, Ringer et al. 2015) on proteins such as cadherin (Conway, Breckenridge et al. 2013), vinculin (Grashoff, Hoffman et al. 2010, Leerberg, Gomez et al. 2014), and talin (Austen, Ringer et al. 2015, Kumar, Ouyang et al. 2016). Quantitative measures of tension may be complemented by qualitative indicators such as nuclear translocation of YAP (Aragona, Panciera et al. 2013), an endogenous mechanosensor, that can be used as a proxy to assess tension experienced by a cell. Combining quantitative biomechanical analysis with qualitative tension sensors and molecular genetic approaches can expose the relationship between tension and MET.

4.2 RESULTS

4.2.1 Surface cells on embryonic cell aggregates transition from mesenchymal to epithelial

To gain insight into the cellular behaviors of deep mesenchymal tissues *in vitro*, we assembled aggregates composed of mesenchymal cells isolated from early gastrula embryos (stage 10; Figure 23A and B) and monitored their phenotypic changes. Briefly isolated mesenchymal cells aggregate within an hour, and then progressively form a compact spherical shape. Using immunostaining and histology to characterize cell phenotypes, cells at early stages of aggregation exhibit many F-actin rich protrusions and assemble fibronectin fibrils at the aggregate surface (Figure 23C). As the aggregate becomes more compact and round (~ 5 hpa), patches of cells on the surface change their phenotype and redistribute fibronectin fibrils leaving distinct patches bare of fibrils. Cells in these patches adopt epithelial-like shapes with distinctive boundaries marked by dense F-actin cables, reduced protrusions, and are cleared of FN fibrils (Figure 23C, see arrows). By 24 hpa, the entire surface develops into a mature epidermis consisting of a tightly packed epithelium including multi-ciliated cells indicated by dense apical actin (Figure 23C) and a complete absence of FN fibrils. Additional experiments were performed to confirm that there was no contamination of endogenous epithelial cells from microsurgery, no confounding effects of cell culture media and no effect of initial aggregate size(Kim, Jackson et al. submitted). Thus, MET is robust and rapid within 3D mesenchymal aggregates.

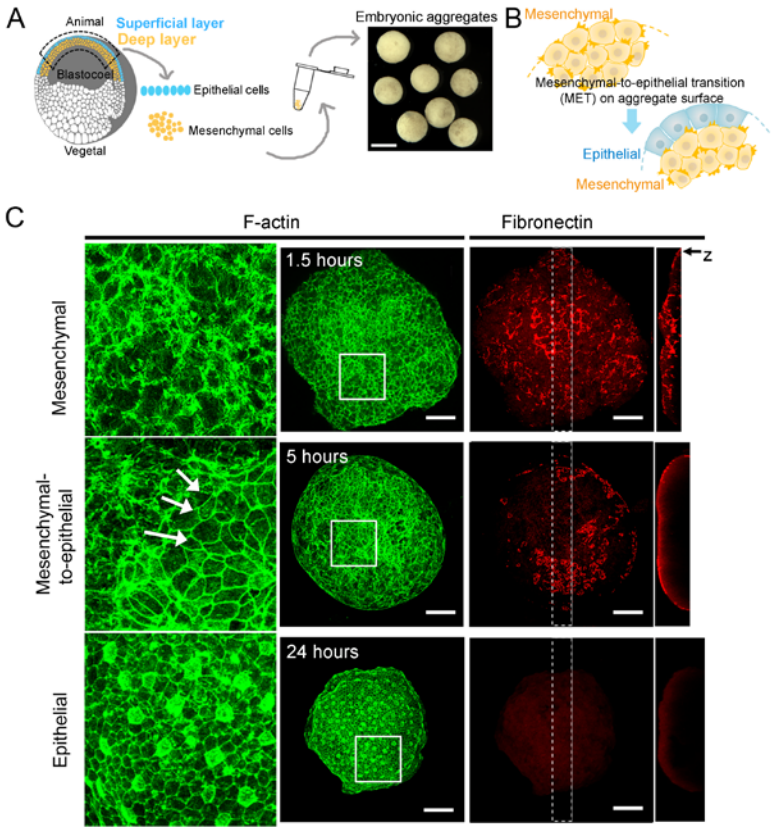

Figure 24. MET within embryonic cell aggregates

(A) Schematic assembly of mesenchymal deep cell aggregates from early Xenopus embryo (stage 10). Scale bar 1 mm. (B) MET progression on aggregate surface. (C) Surface F-actin and fibronectin (FN) from maximum intensity projections at 1.5, 5, and 24 hpa. White inset area of F-actin panel enlarged on left. XZ projection (z, dotted inset) on right shows loss of FN from surface. Arrows indicate epithelial phenotype. Scale bar, 100 µm. Credit = Hye Young Kim.

Transcriptional changes in the mesenchymal aggregate parallel the changes observed in polarity and differentiation factors, including observed localization of aPKC, ZO-1 and expression of cytokeratin and the goblet cell marker interleukin-1 (Itln1; (Kim, Jackson et al. submitted)). We measured transcriptional levels of genes specific to mesenchyme (FN, VimA, and Snail) and epithelia (Cdh1, ZO-1, Krt12. Itln1) in whole embryo, animal cap (both mesenchymal and epithelial), deep ectoderm (mesenchymal), and superficial ectoderm

(epithelial) at before and after MET arise in mesenchymal aggregates (3 and 24 hpa). Immediately after isolation deep ectoderm cells express mesenchymal markers, including vimentin and Snail, while superficial cells of the ectoderm do not (Figure 2E; Expression profile 3 hpa). Between 3 hours and 24 hpa, deep ectoderm cell aggregates downregulate mesenchymal marker genes vimentin and fibronectin while they increase expression of epithelial marker genes E-cadherin (Cdh1), ZO-1, keratin (Krt12) and intelectin 1 (Itln1) (Figure 2F; Fold change). Thus, progressive phenotypic changes from mesenchymal to epithelial cell phenotype at the surface of mesenchymal aggregates represents a novel case of de-novo MET.

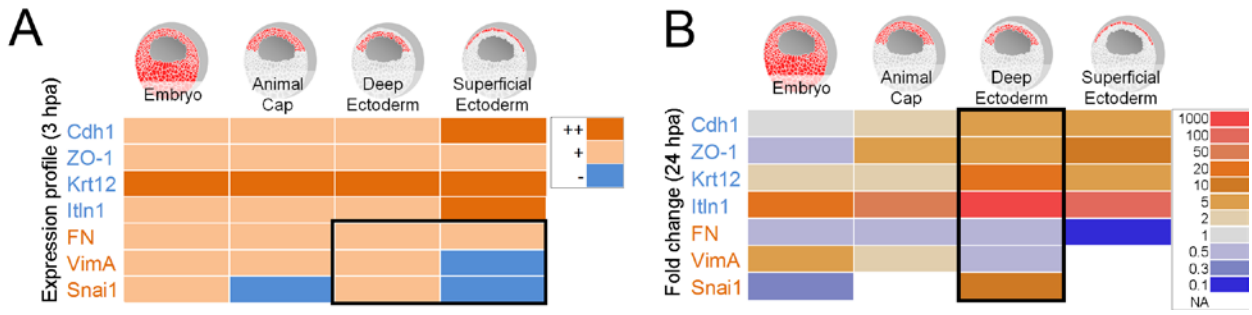

Figure 25. Transcriptional profiling of MET in embryonic cell aggregates

(A) qPCR expression profiling in mesenchymal aggregates at 3 hours post aggregation (hpa) of epithelial (Cdh1, ZO-1, Krt12, and Itln1) and mesenchymal (FN, VimA, and Snai1) genes (top; ++, +, and -) and (B) C_T-based fold changes between 3 hpa to 24 hpa.

4.2.2 Live imaging reveals MET initiation and dynamic MET propagation across the surface of embryonic cell aggregates

Patchy localization of ZO-1 at the surface of aggregates suggested MET might emerge with a specific pattern. To describe the emergence of MET, we tracked the dynamics of ZO-1

(GFP-ZO-1) within live aggregates (Fig. 26A). ZO-1 localizes exclusively at the apical face of epithelial cells and does not localize in *Xenopus* mesenchymal cells (Fesenko, Kurth et al. 2000). After 2.5 hours of imaging, we find the first initiation point where a single cell transitions from being initially protrusive to localizing ZO-1 on its cell boundaries (Fig. 26B). That cell initiates contact with surrounding cells, while expanding its apical surface and ultimately dividing, in order to propagate MET (Fig. 26B). We also observe cells intercalate in from deeper layers to join the growing epithelium (Fig. 26C) as well as adjacent epithelial patches reaching out to merge with each other (Fig. 26D). These live imaging results support our initial findings that MET, i.e., the number of ZO-1 positive cells at the aggregate surface, increases over time, albeit in a stochastic manner. Observations that MET initiates and spreads by involving mesenchymal cells already on the surface of the aggregate suggests that mechanical conditions at the surface of the aggregate can drive MET.

Figure 26. Live imaging reveals various methods for METspreading across surface of aggregate

(A) Individual maximum projection frames from a 3D confocal time-lapse sequence collected from a sector of a mesenchymal cell aggregate. (B) Higher resolution panels of various methods of epithelial expansion below, with (I) RFP-ZO-1 expression and (II) cell outlines drawn and overlaid onto images. The middle panel (blue inset) shows progression of MET within individual cells (blue and green asterisks). A single cell (blue asterisk) is initially protrusive (blue arrows) but assembles ZO-1 at sharp cell-cell boundaries within 2.5 hours. Nearby cells (green asterisks) also assemble ZO-1 and join the first cell in an epithelial patch over the course of an hour. (C-D) The lower panel shows single frames over the later stages (orange inset - single cell MET, and purple inset - patch fusion) as cells continue to intercalate into the growing epithelium (orange arrow) and larger patches of epithelial cells merge (purple). (A-D) All scale bars are 50 μm.

4.2.3 Emergence of YAP nuclear localization suggests tension arises stochastically on the aggregate surface

The stochastic emergence of MET across the surface of the aggregate suggested that cells may be responding to local variation in tension. While a number of FRET-based tension sensors have been utilized to quantify tension experienced by cells (Cost, Ringer et al. 2015), Yes-associated protein 1 (YAP) translocation to the cell nucleus may also serve as a qualitative estimate of tension patterns throughout a tissue (Chanet and Martin 2014). After expressing YAP-GFP in mesenchymal cell aggregates we found stochastic patterns of YAP nuclear localization (Figure 27A and A'). If operating independently from its function in the Hippo signaling pathway, YAP can move into the nucleus in response to mechanical forces or stress fiber formation (Chanet and Martin 2014), such as cell shape changes and ECM rigidity (Dupont, Morsut et al. 2011), and stretch (Aragona, Panciera et al. 2013). To determine whether YAP nuclear localization reflected local tension we compared aggregates at 2 hpa (not shown), 5 hpa (Figure 27A and A'), and blebbistatin-treated aggregates at 5 hpa (Fig. 27B and B') and observed more nuclear localization of YAP in 5 hpa (Fig. 27A; arrows) and reduced nuclear localization after incubation with blebbistatin (Figure 27B and B'). Furthermore, YAP localization peaks appear stochastically across the surface of the aggregate (Figure 27A’, arrows). To quantify YAP nuclear localization, we counterstained with Hoechst 33342, segmented the images to identify cell nuclei, and measured the intensity ratio of YAP in the cell nucleus compared to the cytoplasm (Figure 27C). Using this method, we find cells in 5 hpa aggregates have significantly higher nuclear localization than cells in 2 hpa controls and 5 hpa blebbistatin-treated aggregates (Figure 27D) suggesting these cells experience higher tension. Using ZO-1 to identify cells that have undergone MET, we also find that cells that have epithelialized have significantly higher

YAP nuclear localization than mesenchymal cells (Figure 27E). YAP nuclear localization appears to peak within or near cells undergoing MET (Figure 27F), suggesting that tension may serve as a trigger or cue for *de novo* MET.

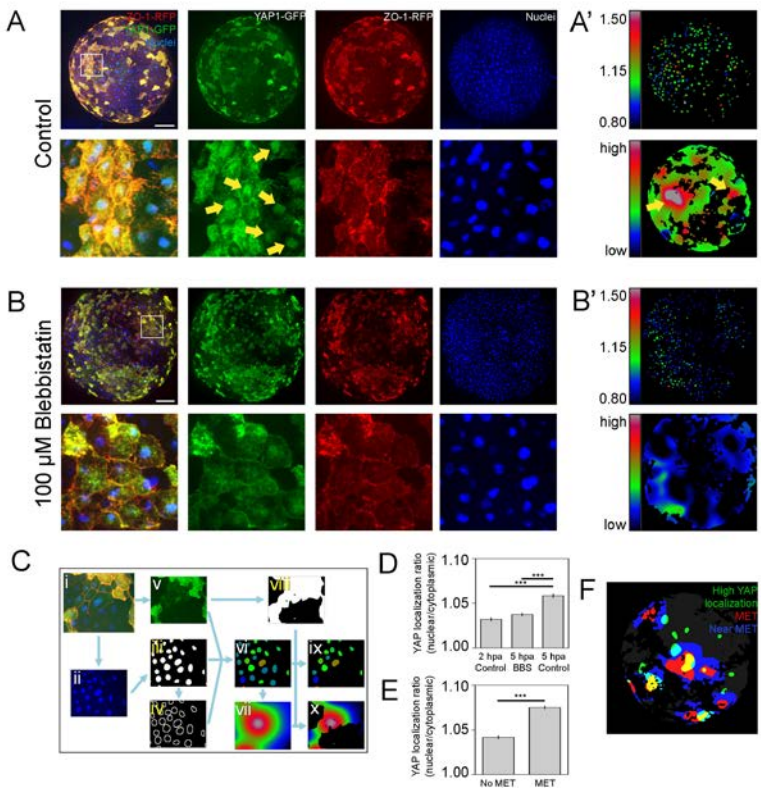

Figure 27. Cells undergoing MET have greater YAP nuclear localization.

(A - B) Maximum intensity projections of (A) control and (B) blebbistatin-treated aggregates at 5 hpa. Aggregates express fluorescently-tagged YAP-GFP (green) and ZO-1-RFP (red) and were stained with Hoechst 33342 (blue). Scale bar, 100 µm. (A' - B') YAP localization ratios of each nucleus were used to generate tension maps. (C) Workflow of image analysis. The original image (i) is split into channels. The Hoechst 33342 channel (ii) is segmented to generate a nuclei mask (iii), which is then dilated and subtracted to obtain a cytoplasm mask (iv). Using the YAP channel (v), the mean intensity for each cell nucleus and cytoplasm region is calculated and divided to get the raw YAP nuclear localization ratio image (vi), which is passed through a heavy Gaussian blur to obtain a tension map (vii). Finally, using the YAP expression to generate a mask (viii), non-expressing cells are excluded from the final localization ratio image (ix) and tension map (x). (D) YAP localization ratios of 5 hpa aggregates (N = 2040 nuclei from 7 aggregates) are higher than blebbistatin treated aggregates (N = 3471 nuclei from 7 aggregates) and 2 hpa aggregates (N = 2167 nuclei from 7 aggregates). (E) Cells that undergo MET (N = 7789 nuclei from 21 aggregates) have higher YAP localization than mesenchymal cells (N = 284 nuclei from 21 aggregates). (F) Regions of YAP nuclear localization (green) appears within cells that have undergone MET (red) or within 50 microns of cells that have undergone MET (blue).

4.2.4 Macroscopic stiffening of multicellular aggregate correlates with MET progression

Since the EMT during cancer can be triggered by stiffening of the stromal microenvironment (Levental, Yu et al. 2009) and YAP translocates to the nucleus near the site of *de novo* MET we hypothesized that MET might occur in response to mechanical changes in the microenvironment. To investigate this, we measured the creep compliance near the surface of aggregates (to a depth of ~ 125 μm) using microaspiration (von Dassow, Strother et al. 2010). Negative pressure (-10.1 Pa) applied through a microchannel to a small patch on the surface of the aggregate reports the mechanical compliance based on the degree of tissue movement into the channel (Figure 28A-C). Aggregates at 3, 6, and 12 hpa, respectively, before, during, and after MET revealed tissue compliance significantly decreases over the first 6 hours as MET emerges and continued to decrease at later phases of MET (Figure 28D). Comparing early aggregates (3 hpa) either with or without superficial ectoderm cells revealed mechanical properties of epithelial and mesenchymal tissues are not significantly different (Figure 28E), suggesting that the decreases in tissue compliance was not merely a consequence of epithelialization. Stiffening of the aggregate over the early phases of aggregation and during MET suggested that cell contractility or changes to cell adhesions trigger MET on the aggregate surface.

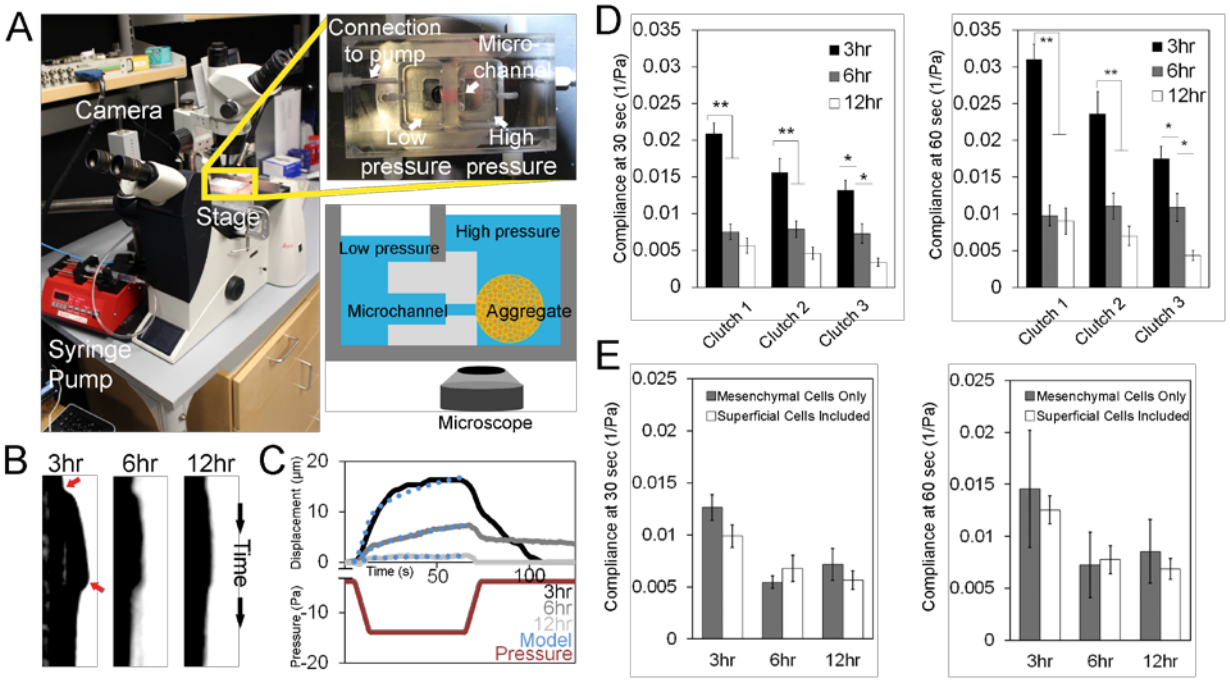

Figure 28. MET progression coincides with reduced tissue compliance.

(A) Photograph of customized micro-aspiration rig, with zoom-in of the chamber and schematic of micro-aspirator used to measure surface tension of aggregates by adjusting liquid pressure of the chamber. (B) Representative kymographs of tissue displacement over the length of the microaspiration experiments at 3, 6 and 12 hpa. Red arrows indicate when suction pressure is applied and then released. (C) Representative graph of aspirated distance of an aggregate at 3 (black), 6 (dark gray), 12 hpa (light gray) relative to the pressure applied (red). The power-law model for creep compliance (blue) was fit to this data. (D) Creep compliance at 30 and 60 seconds indicating steady-state from micro-aspiration at 3 (black), 6 (gray) and 12 hpa (white). 11 to16 aggregates were measured for each time point and repeated over three clutches (three clutches; * $P<0.05$, ** $P<0.01$, Error bar = +/- SE). (E) Creep compliance at 30 and 60 seconds from micro-aspiration of aggregates consisting of only mesenchymal cells (gray) or both mesenchymal and superficial epithelial cells (white) 3, 6, and 12 hours post aggregation.

4.2.5 MET can be delayed or accelerated by modulating tissue mechanics in embryonic cell aggregates.

To test whether cellular processes control MET through mechanics, we modulated actomyosin contractility and tracked the progression of MET. Because we observed that reducing cell-mediated tension inhibited MET (Kim, Jackson et al. submitted), we wanted to test whether increasing cell contractility might accelerate MET. To increase contractility we incubated aggregates with Calyculin A (10 nM), a myosin light chain phosphatase inhibitor which increases contractility (Yam, Wilson et al. 2007, Kim and Davidson 2011) and also expressed a potent activator of myosin contractility, the constitutively active form of arhgef2 (arhgef2^{C55R} ; alternatively RhoGEF2, GEF-H1, or XLfc (Kwan and Kirschner 2005)), a RhoA-specific guanine nucleotide exchange factor known to strongly lower tissue compliance (Zhou, Kim et al. 2010). Both treatments significantly increased MET (1.6-fold by Calyculin A and 4.6-fold by arhgef2^{C55R}; Figure 29A and C). Incubation of aggregates in calyculin A drives early co-localization of ZO-1 with cortical F-actin in newly-epithelialized cells (Figure 29B). The effects of arhgef2^{C55R} can be directly attributed to cell contractility, since induced MET can be completely abolished by incubation with Y27632 (Figure 30A and B). We used microaspiration to confirm that Calyculin A and Blebbistatin respectively decreased and increased compliance over control aggregates (Figure 29D) in concordance with their effects on actomyosin contractility. In summary, our findings support the hypothesis that contractility and tissue compliance regulate the onset of MET and increased levels of cell contractility can accelerate progression of MET.

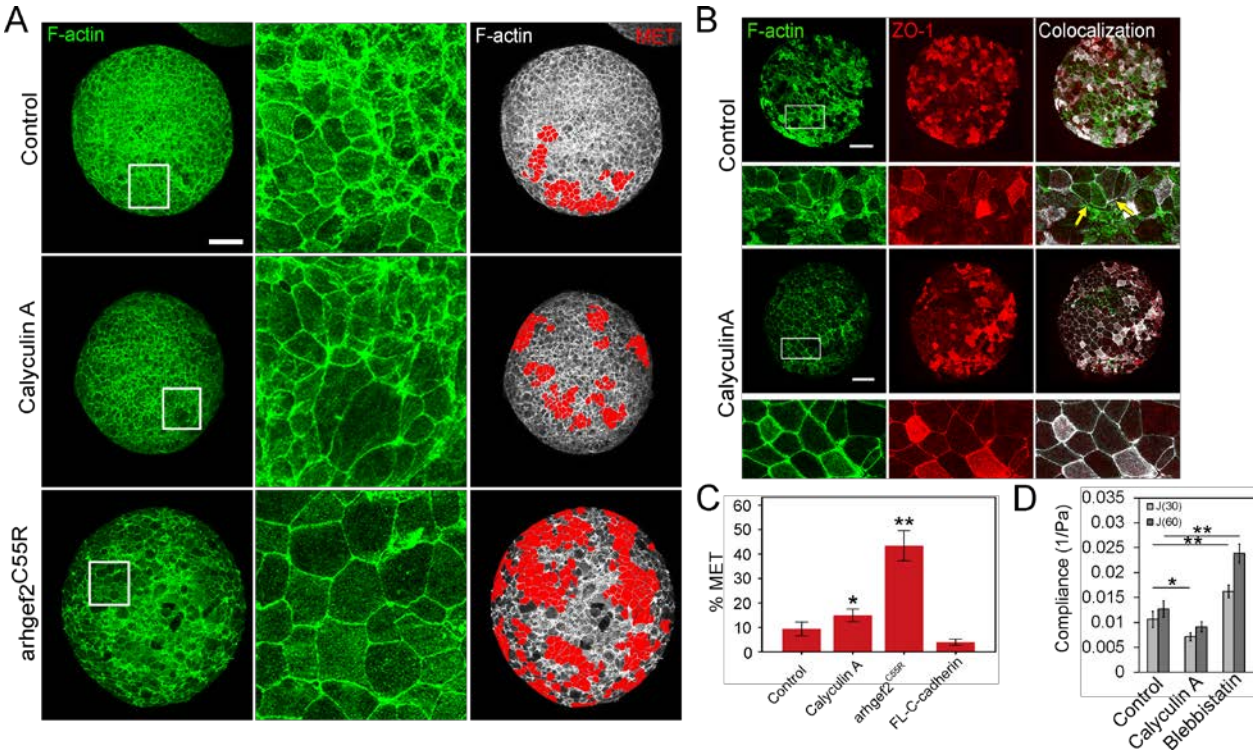

Figure 29. Increased cell contractility and low compliance drives MET.

(A) Maximum intensity projection of F-actin stained aggregates at 5 hpa (Control, n=30) show MET increases with increased cell contractility (Calyculin A, n=12; arhgef2C55R , n=10). Detailed view of white inset shown in right panel. Red filled cells indicate epithelial cells (MET) and their surface area are quantified in a bar graph (B; FL-C-cadherin, n=20; * P<0.05, ** P<0.01; Error bar, +/- SE).(B) Aggregates expressing ZO-1 RFP mRNA (red) and stained for F-actin (green) showing co-localization of tight junctions and cortical actin in post-MET epithelial cells (white) in both control and Calyculin A treated aggregates. Scale bars in (A) and (B) are 100 µm. (D) Creep compliance at 30 and 60 seconds indicating steady-state from micro-aspiration at 6 hpa after 4 hours of small molecule inhibitor treatment. Data represents 4 clutches pooled with 22 to 25 aggregates for each treatment (* P<0.05, ** P<0.01, Error bar = +/- SE). Panels A and C were created by Hye Young Kim.

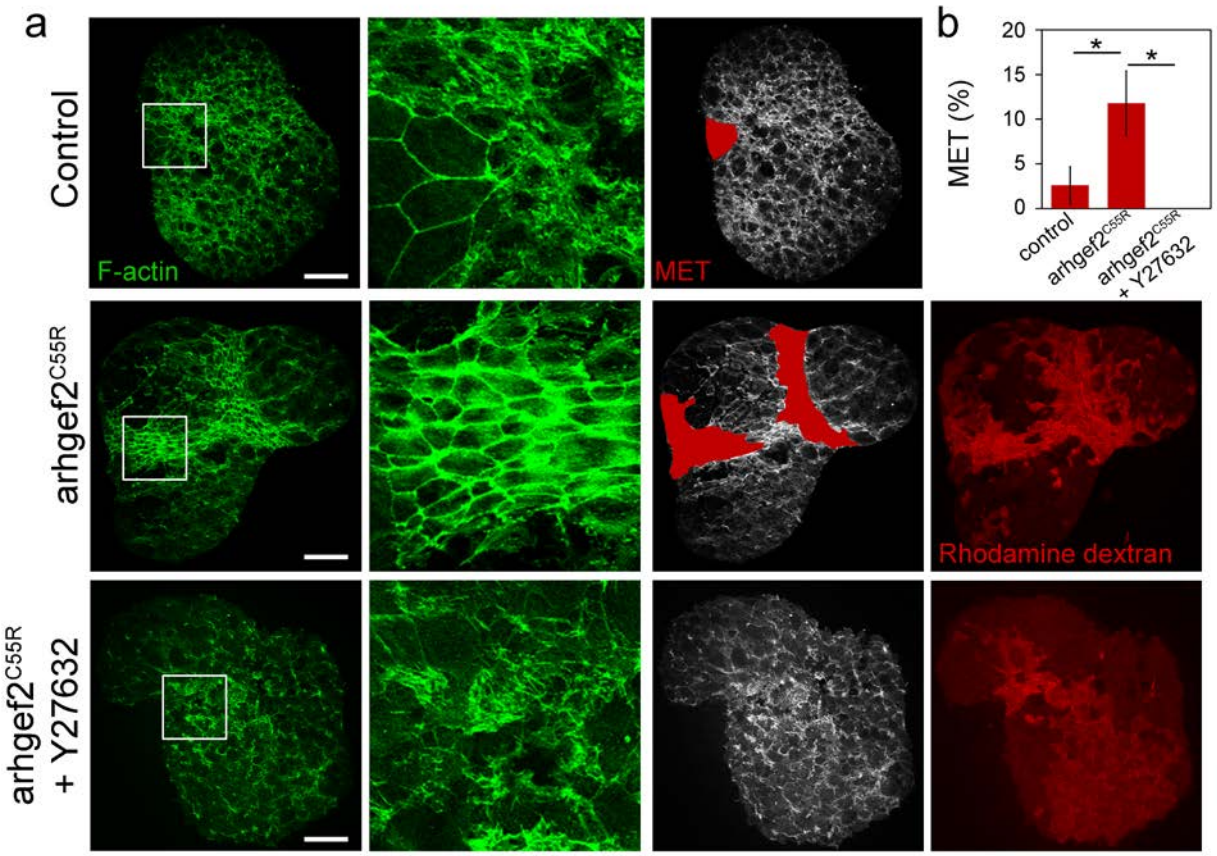

Figure 30. Y27632 inhibits MET-inducing activity of arhgef2C55R.

(A) Z-projections of F-actin to visualize MET (red overlay) while rhodamine dextran co-injected with arhgef2^{C55R} shows where construct is expressed. (B) Percentage of surface area of aggregate having undergone MET demonstrates arhgef2^{C55R} can accelerate MET and Y27632 can reverse those effects.

4.3 DISCUSSION

In this study we introduced a novel 3D model for MET using aggregates composed of embryonic mesenchymal cells from the *Xenopus* deep ectoderm and demonstrated direct mechanical control of the transition of mesenchymal cells to an epithelial phenotype. In the gastrula, deep ectoderm cells are fully mesenchymal, do not localize apical-defining polarity factors such as Crumbs or aPKC and do not assemble apical junctions (Chalmers, Welchman et al. 2002, Chalmers, Pambos et al. 2005, Wang, Cha et al. 2013). Like classical mesenchymal cells, they maintain loose associations as they extend protrusions, migrate, and rearrange beneath the superficial layer (Marsden and DeSimone 2001). Furthermore deep mesenchymal cells isolated from the animal pole ectoderm of early stage embryos exhibit pluripotent potential to differentiate into neural, mesodermal, and endodermal lineages if cultured in isolation (Kuroda, Fuentealba et al. 2005) or supplied appropriate induction factors (Ariizumi and Asashima 2001); the developmental potential of these cells parallel the potential of mammalian embryonic stem cells. Deep ectodermal mesenchymal cells do not normally differentiate as goblet cells, demonstrated by Itln1 expression (Figure 2D, E, and F) and fate mapping of goblet cells of the epidermis (Figure 2C), consistent with previous reports (Dubaissi, Rousseau et al. 2014). Thus, the MET that occurs on the surface of mesenchymal aggregates does not reflect an endogenous program of MET in frog embryogenesis and is distinct from programs of epithelialization in other species such as the differentiation of the zebrafish enveloping layer (Manning, Foty et al. 2010, Krens, Mollmert et al. 2011).

MET occurring on the surface of 3D mesenchymal aggregates allow us to test the hypothesis that MET is not simply a reversal of EMT and to explore the mechanobiology of cell phenotypic transitions. Progressive MET on the surfaces of mesenchymal aggregates enables live imaging with high spatial and temporal resolution and provides access to cell and tissue mechanics, which to our knowledge are not accessible in any other model of MET or epithelialization. Recent cancer biology studies have revealed critical transcriptional differences between EMT and MET (Pattabiraman, Bierie et al. 2016) suggesting an intermediate state in the progression of cells from epithelial to mesenchymal but a direct transition from mesenchymal to epithelial (Lu, Jolly et al. 2013, Zhang, Tian et al. 2014). The role of signaling or mechanical control of these transitions has not yet been addressed. By contrast, there is growing evidence that tissue compliance and anisotropic tension play key roles in tissue self-assembly and coordination of growth during epithelial morphogenesis (Heisenberg and Bellaiche 2013), but few studies on the role of these factors during mesenchymal morphogenesis. The progressive increase in YAP localization (Figure 4D) and decrease of bulk tissue compliance (Figure 5C) as MET spreads across the aggregate surface demonstrates a clear link between tension and the stochastic emergence of MET.

The challenges to investigating the biology of MET, the occurrence of intermediate transcriptional states, or the role of mechanics in controlling MET stem from the limited access to this fundamentally 3D-process. For instance, during the spread of cancer, MET occurs at very low frequencies, deep within the adult body (Nieto, Sargent et al. 1994, Gunasinghe, Wells et al. 2012). Similarly, instances of MET during organogenesis involve small numbers of cells deep within the embryo (Chaffer, Thompson et al. 2007). The rapid and predictable MET on the surface of mesenchymal aggregates appears to mimic *in vivo* processes of MET. Furthermore,

our findings suggest the molecular and physical pathways driving MET in dense *Xenopus* mesenchymal aggregates may model the processes that direct MET during development and organogenesis (Nance 2014, Kim, Jackson et al. 2016) including progressive epithelialization during establishment of cell polarity in *C. elegans* (Nance, Munro et al. 2003) as well as the transition from morula to blastula (Fierro-Gonzalez, White et al. 2013, White and Plachta 2015) and polarization of the inner cell mass in mammals (Morris 2011, Schrode, Xenopoulos et al. 2013).

During development, we propose mesenchymal cells that contribute to epithelia during organogenesis do so by integrating positional information from both biochemical and mechanical cues. To test this hypothesis will require the use of new 3D models of MET and the adoption of new techniques such as sophisticated optogenetic or microfluidic tools for controlling fine-scale mechanical heterogeneity. Experimental control over the mesenchymal tissue microenvironment will enable studies of mechanosensing and mechanotransduction and will likely expose the mechanical cues driving this and other cases of MET and reveal how mechanical cues are integrated with more classical signaling systems.

4.3.1 Limitations

Although our *ex vivo* aggregate system provides unprecedented access to dissect the molecular signaling pathway and tissue mechanics that contribute to MET, that access may ultimately sacrifice some of the complexity that can only be resolved through studying MET *in vivo*. Furthermore, we have currently only applied this 3D cell aggregate to *Xenopus* animal cap cells and therefore can only study one type of MET responsible for regenerating goblet cells. However, it will be interesting to see if other mesenchymal cell types can be assembled into an

aggregate in a similar manner and whether these can be coaxed by mechanical cues into epithelial cell types.

It should also be noted that live-imaging of MET in our aggregate system is not without its technical difficulties. The assembly of tight junction appears to be particularly sensitive to phototoxicity from confocal microscopy. Therefore, it was necessary to limit the exposure of aggregates to laser emission, through reducing the acquisition rate and laser intensity. Aggregates are also highly sensitive to mechanical constraints which can easily inhibit MET. Live imaging of MET requires limited physical constraints that do not perturb conditions within the aggregate.

4.4 EXPERIMENTAL PROCEDURES

4.4.1 Embryonic cell aggregates

To assemble the mesenchymal cell aggregates, we first microsurgically isolated animal cap explants from early gastrula embryos (stage 10) in Danilchik's For Amy (DFA) medium supplemented with antibiotic and antimycotic (Sigma). Next, the superficial layer of the animal cap explant was separated from deep mesenchymal cells in Calcium/Magnesium free-DFA and discarded. Freshly isolated deep mesenchymal cells were transferred to a PCR tube using a pipette. To provide a non-adherent surface for aggregation, PCR tubes were coated with 1% BSA overnight at 4°C. To confirm that our methods did not produce aggregates contaminated by epithelial cells from the superficial layer, the surface layer of whole embryos were labeled with NHS-Rhodamine (Thermo Scientific)(Edlund, Davidson et al. 2013) and used to assemble

mesenchymal aggregates (Figure 1D and Figure S1A). Two animal cap explants were used to assemble a single mesenchymal aggregate except to make different sizes of aggregates (Figure 1E). Mesenchymal aggregates acquire a spherical shape within 2 to 3 hours post aggregation (hpa), at which point they can be subjected to further procedures including treatment with pharmacological inhibitors, measurement of surface tension, and live imaging.

4.4.2 Microinjection

To visualize or overexpress proteins, mRNA of the desired protein was injected at the 1- or 2-cell stage of fertilized embryos. GFP-ZO-1 was constructed by moving the entire eGFP-human TJP1 fusion protein obtained from Addgene (Plasmid #30313; Addgene, Cambridge, MA) into pCS2+, mRFP-ZO-1 plasmid provided kindly by Dr. Anne Miller. To induce cell contractility, embryos were injected with arhgef2^{C55R} at ~87.5 pg per embryo (Zhou, Kim et al. 2010). To alter cell-cell adhesion, ΔE-C-cadherin (~1ng per embryo), ΔC-C-cadherin (~3.8ng per embryo), and FL-C-cadherin (~1.4 ng per embryo) were injected and ectopic gene expression was confirmed by both myc staining and phenotype (Kurth, Fesenko et al. 1999).

4.4.3 Histology and immunofluorescence

Aggregates were fixed with 4% paraformaldehyde and 0.25% glutaraldehyde in PBST (1x PBS with 0.1% Triton X-100) for 15 minutes at room temperature for visualizing F-actin and fibronectin. For immunofluorescence of ZO-1, keratin, aPKC and Itln1, aggregates were fixed with ice cold Dent's fix (4:1 of Methanol: DMSO). Fixed aggregates were washed for 30 minutes with PBST and blocked with 10% goat serum in PBST for 1 hour prior to antibody staining.

Primary antibodies for FN (4H2), aPKC (nPKCζ (C-20) sc-216; Santa Cruz), acetylated tubulin (clone 6-11B-1; Sigma), ZO-1 (Invitrogen), keratin (1h5; Developmental Studies Hybridoma Bank), Myc (9E10; Millipore), and Itln1 (gift from Dr. Eamon Dubaissi) were used with 1:200 dilution and incubated overnight on a nutator at 4°C. After washing, the samples were incubated with the appropriate secondary antibody at a 1:200 dilution overnight at 4° C. F-actin and cell nuclei were visualized with BODIPY-FL–phallacidin (1:800) and either YO-PRO (1:10000, Invitrogen) or Hoechst 33342 (1:2000, Thermo Fisher), respectively. Cells were evaluated as having undergone MET if they exhibited no F-actin protrusions and had strong F-actin localization to cell boundaries, which is indicative of co-localization with ZO-1.

For YAP nuclear localization studies, aggregates expressing YAP-GFP and ZO-1-RFP were fixed using 4% PFA for 15 minutes. For analysis of the images, tiled images were stitched together using an ImageJ plugin (Preibisch, Saalfeld et al. 2009) and max-projected. The Hoescht channel was used to segment nuclei using a simple binary threshold. The YAP-GFP channel was used to segment and exclude cells not expressing fluorescent proteins. A custom macro was written to quantify the mean nuclear intensity of YAP-GFP expression of each cell and then by dilating the nuclear masks, calculating the mean cytoplasmic intensity. YAP localization ratios (nuclear to cytoplasmic) are shown in Fig. 4A'-B'. YAP nuclear localization maps were created using a Gaussian blur with a large kernel size to visualize neighborhood averages of YAP localization ratios. A schematic of the image analysis sequence is shown in Figure 4C.

4.4.4 Compliance measurements using micro-aspiration

To quantify tissue mechanics, aggregates were placed in the high-pressure reservoir of a dual-reservoir micro-aspirator apparatus (von Dassow and Davidson 2009, von Dassow, Strother

et al. 2010). Aggregates were gently compressed using hair tools against a 125 μm diameter channel cast across a PDMS block to form a seal across the opening of the channel. A pressure differential at the channel opening was controlled hydrostatically using a computer-controlled syringe pump. After calibrating the channel to obtain zero flow, a small baseline pressure of approximately -3.8 Pa was applied in order to initiate suction and seal the aggregate onto the channel opening. The tissue aggregate was allowed to relax for 5 minutes before the 125-second micro-aspiration protocol was initiated. At the start of the protocol, the aggregate was imaged at 1 frame per second to obtain an initial aspiration length for the tissue. After 5 frames, the syringe pump removed 800 μL of media from the low-pressure reservoir resulting in a -10 Pa suction pressure applied to the aggregate. After 60 seconds of -10 Pa suction, the 800 μ L was media was replaced to return to the baseline pressure and imaging was continued for an additional 60 seconds to confirm the aggregate was properly sealed onto the channel opening. When carrying out repeated measurements on the same sample, the aggregate was rotated to a different position. The tissue boundary was tracked either manually or automatically with Canney-Deriche edge detection (Schneider, Rasband et al. 2012). The aspirated length was measured and the data was fit to the power-law model for creep compliance, which has previously been shown to provide an adequate fit for embryonic tissue responses to stress-application (von Dassow, Strother et al. 2010). Time-dependent compliance at 30 seconds (fast response) and 60 seconds (steady-state) is reported as the measurement of aggregate mechanical properties. Compliance calculations for each sample are independent of the magnitude of the pressure applied (von Dassow, Strother et al. 2010). The suction force of -10 Pa was chosen in order to sufficiently displace aggregate tissues within the resolution of our imaging system without surpassing the critical pressure, thus

ensuring measurements are representative of tissue elastic properties (Sato, Levesque et al. 1987).

4.4.5 qPCR

Gene expression was determined by standard qPCR methods. To evaluate epithelial and mesenchymal gene expression in embryonic tissues, RNA was isolated using the Quick-RNA MiniPrep kit (Zymo Research) with 10 samples each of animal caps, deep cell aggregates and superficial cell aggregates and 5 whole embryos at 3, 6 and 24 hours post aggregation. RNA concentrations were measured using the NanoDrop and total RNA was used to create cDNA with the AccuScript High Fidelity 1st Strand cDNA Synthesis Kit (Agilent Technologies, Inc.). Forward and reverse primers (see Table 1) were designed for *Xenopus laevis* epithelial genes ZO-1, Cdh1, Itln1 and Krt12, mesenchymal genes FN, VimA and Snai1 and controls H4A. qPCR protocols were designed and carried out using the Bio-Rad iQ5 thermal cycler and PCR detection system. 10 uL reaction volumes were prepared according to SsoAdvanced Universal SYBR Green Supermix (Bio-Rad) instruction manual. 40 amplification cycles were carried out followed by a melting curve analysis to ensure product purity. Categorization of gene expression at 3 hpa was done using C_T values (C_T > 30 –, 25-30 +, < 25 ++). Fold changes at 24 hpa were calculated using the Livak method, using H4 as the reference gene and 3 hpa as the calibrator (Šindelka, Ferjentsik et al. 2006). Results are consistent across triplicate experiments with two replicates.

Table 1. qPCR Primers for aggregate MET studies

0	Alternate Name	GenBase Name	Accession	Forward Primer	Reverse Primer
Cdh1	E-Cadherin	Cdh1.S	XB-GENE-6464305	GCTGTTGTTGCTCTTACT	CGAGTCTCATCTTCTGGA
ZO-1	TJP-1	Tjp-1.L	XB-GENE-17332215	ATATCCAAGCAGTCAGAGA	TCATCTTCATCATCATCTTCC
Krt12		Krt12.S	XB-GENE-17333623	TTCCACATCACAATCATCTT	ACCACTTCTTCCACGATA
Itln1	Xeel	Itln1.L	XB-GENE-6256033	ACTGAGAGGGCTACACTTGCT	ATGTAACCACCTCCTCCAATGC
FN		Fn1.S	XB-GENE-865084	GGTGGAGGTGTGACAATT	TTGGTATCTCTGTGTAAACTGA
VimA		Vim.L	XB-GENE-866225	GCTAATCGCAACAATGATG	TTGAATAGTGTCCTGATAGTTAG
Snai1		Snai1.L	XB-GENE-865328	GGAGAGTCAGACAGTGTATA	CAAGAGGTGTGTAGTAAGC
H4		hist1h4a.L	XB-GENE-6493984	GACGCTGTCACCTACACCGAG	CGCCGAAGCCGTAGAGAGTG

4.4.6 Statistical Analysis

Statistical differences of creep compliance among different time points post aggregation was calculated with ANOVA and planned repeated contrast. ANOVA of drug treated aggregates showed data did not vary significantly by clutch ($P > 0.05$), so data was pooled together. Statistical differences between drug treatments were calculated with ANOVA and planned simple contrast. Significance of percentages of MET between control and different experimental conditions were calculated using Mann-Whitney U-test (IBM SPSS, version 22).

5.0 MECHANOSENSORY DEFECTS PERTURB MESENCHYMAL-TO-EPITHELIAL TRANSITION IN HEART PROGENITOR CELLS

The following chapter addresses Aim 3 to investigate the role of mechanics and MET in the mutant protein SHP-2^{N308D}, which is associated with Noonan Syndrome (NS) related CHDs. Experiments for this manuscript are currently ongoing. In brief, we demonstrate a capability of recreating the NS cardiac phenotype in *Xenopus* and show that overexpression of SHP-2 causes a 30% increase in bulk tissue compliance within the heart-forming region. We also find that MET occurs abnormally in HPCs expressing SHP-2^{N308D}, i.e. decreased aPKC localization and upregulated ZO-1. Using mesenchymal cell aggregates, which provides finer access to molecular techniques like qPCR, we find that SHP-2^{N308D} causes a cell-autonomous MET and downregulates aPKC transcription, which suggested that SHP-2^{N308D} causes a mechanotransduction defect in HPCs. After confirming fibrillar fibronectin as a requirement to transmit mechanical information between the endoderm and HPCs, we see that the SHP-2^{N308D} induced abnormal MET persists when fibrillar fibronectin is knocked down. Therefore, we believe SHP-2^{N308D} is causing errors in early heart development by driving abnormal MET to perturb heart field organization, leading to the cascade of development defects that ultimately cause the observed CHD phenotype.

5.1 INTRODUCTION

Congenital heart defects (CHDs) are the leading cause of death in infants in the U.S. (Nembhard, Wang et al. 2010) and yet many of the physical mechanisms driving heart development remain poorly understood. In early heart development, just after gastrulation, bilateral fields of heart progenitor cells (HPCs) move from the anterior lateral plate mesoderm to the ventral midline. Halfway through this process, HPCs undergo a mesenchymal-to-epithelial transition (MET) regulated by temporally controlled mechanical cues (Jackson, Kim et al. In review). Correct timing of this MET appears crucial for proper heart structure and function, which suggests that the developmental errors that lead to certain CHDs may arise due to abnormal MET. Following arrival within the ventral heart forming region (HFR), HPC epithelia fold into a heart tube, which then undergoes looping, septation and trabeculation to shape the larval heart (Warkman and Krieg 2007).

During the process of MET, cells establish apicobasal polarity, which includes the assembly of tight junctions, which are composed of the protein ZO-1 and connected to the actin cytoskeleton via α-catenin. Prior to polarization, ZO-1 and α-catenin associate in the cytosol to form immature tight junction complexes. Once the Par6/Par3/aPKC complex is activated, often through Cdc42 signaling, mature junctions are assembled by actomyosin contractility induced by further RhoGTPase signaling (Zihni, Mills et al. 2016). Numerous developmental defects have been connected to errors in apicobasal polarity establishment, including neural tube defects due to Par3 mutations (Chen, An et al. 2016) and cardiac defects due to errors in the regulation of mechanics {Jackson, In review #213}

In order to initiate MET, HPCs must have a framework capable of probing and receiving mechanical cues from their environment. Cells use extracellular matrix (ECM) proteins to obtain

information to guide cell differentiation (Engler, Sen et al. 2006), migration and actin organization (Trichet, Le Digabel et al. 2012). Focal adhesion sites are thought to mediate mechanotransduction as they connect the ECM to the cytoskeleton (Geiger, Spatz et al. 2009) and one of the key members of focal adhesions is the integrin heterodimer, consisting of an α- and β-subunit. In *Xenopus*, β1-integrin and α5 are ubiquitiously expressed throughout the early stages of heart development (Ransom, Hens et al. 1993, Joos, Whittaker et al. 1995) and we have previously shown their role in regulating cell behaviors during convergent extension (Davidson, Marsden et al. 2006). In particular, β1-integrin is shown to stimulate downstream effectors such as RhoGEFs in response to a mechanical stress (Guilluy, Swaminathan et al. 2011), which in turn remodel the cytoskeleton to permit the cell to adapt to mechanical changes. Computational modeling has also demonstrated β1-integrin could transition between its active and inactive forms in response to mechanical stress, further implicating its role in mechanotransduction (Puklin-Faucher, Gao et al. 2006).

One regulator of integrin signaling is the ubiquitous protein phosphatase SHP-2. SHP-2 associates with integrin rafts to regulate cell adhesions to ECM, FAK phosphorylation and Rho signaling (Inagaki, Noguchi et al. 2000, Lacalle, Mira et al. 2002). In terms of CHDs, mutations in SHP-2 are associated with Noonan Syndrome (NS), one of the most common CHDs (Sharland, Burch et al. 1992, Noonan 1994). An estimated 4000 babies are born with NS annually, half of which are associated with point mutations in the protein phosphatase SHP-2, the most common being SHP-2^{N308D} (Tartaglia, Mehler et al. 2001). Previous work has demonstrated the ability to recreate SHP-2^{N308D} induced CHDs in frog, which was ultimately attributed to decreased HPC proliferation, upregulated ROCK and perturbed myocardial cytoskeletal dynamics at stages post-heart tube formation (Langdon, Tandon et al. 2012). As early defects,

e.g. during stages of HPC migration to the midline and MET, can cause a cascade of errors in later development (Jackson, Kim et al. In review), we wondered if SHP-2^{N308D} induced CHDs may be the result of early defects in HPC MET. SHP-2 is known to influence cell phenotypic changes, including driving EMTs in cancer (Zhang, Zhao et al. 2016) while Rho/ROCK signaling is a known requirement of tight junction formation (Gopalakrishnan, Raman et al. 1998, Walsh, Hopkins et al. 2001, Kwon, Kim et al. 2016). This led us to suspect that SHP-2^{N308D} may be perturbing MET in HPCs.

Here, we recreated the NS phenotype of SHP-2^{N308D} induced CHDs in *Xenopus* embryos and investigated its effects on MET and tissue mechanics in HPCs. Using microaspiration, we find that overexpression of wildtype and mutant SHP-2 proteins causes an increase in tissue compliance. Contrary to our previous model that higher compliance would result in less MET, this causes a cell-autonomous abnormal MET. By first identifying fibrillar fibronectin and β1-integrin as key players in the HPC mechanosensory pathway, we find that SHP-2^{N308D} overrides the native mechanosensory framework to upregulate tight junction formation in early heart development.

5.2 RESULTS

5.2.1 SHP-2 overexpression changes bulk mechanical properties of the HFR and causes developmental defects

Because SHP-2^{N308D} was shown to upregulate ROCK and initiate remodeling of the actin cytoskeleton (Langdon, Tandon et al. 2012), we hypothesized that overexpressing SHP-2^{N308D} would perturb bulk tissue mechanics within the HFR. As we would expect, overexpressing both mutant and wildtype SHP-2 caused whole embryo defects (Fig. 31A), including a reduced anteroposterior length (Fig. 31B) and high rate of cardiac edemas (Fig. 31C). We used microaspiration to measure the bulk tissue compliance within the HFR as done previously (von Dassow and Davidson 2009, von Dassow, Strother et al. 2010, Jackson, Kim et al. In review, Kim, Jackson et al. submitted).

Interestingly, overexpressing both mutant and wildtype SHP-2 causes an approximately 30% increase in bulk tissue compliance within the HFR at Stage 22 (Fig. 31E), which is just prior to MET initiation (Jackson, Kim et al. In review). Culturing embryos out to post-looping stages (Stage 37), we find cardiac defects in both SHP-2^{FL} and SHP-2^{N308D} injected embryos, although SHP-2^{FL} appear reduced in size while SHP-2^{N308D} hearts appear hypertrophic, a common presentation of NS (Sharland, Burch et al. 1992). Thus, we conclude that both mutant and wildtype SHP-2 affects bulk tissue mechanical properties within the HFR and causes cardiac defects, which led us to wonder if the two phenomena are connected.

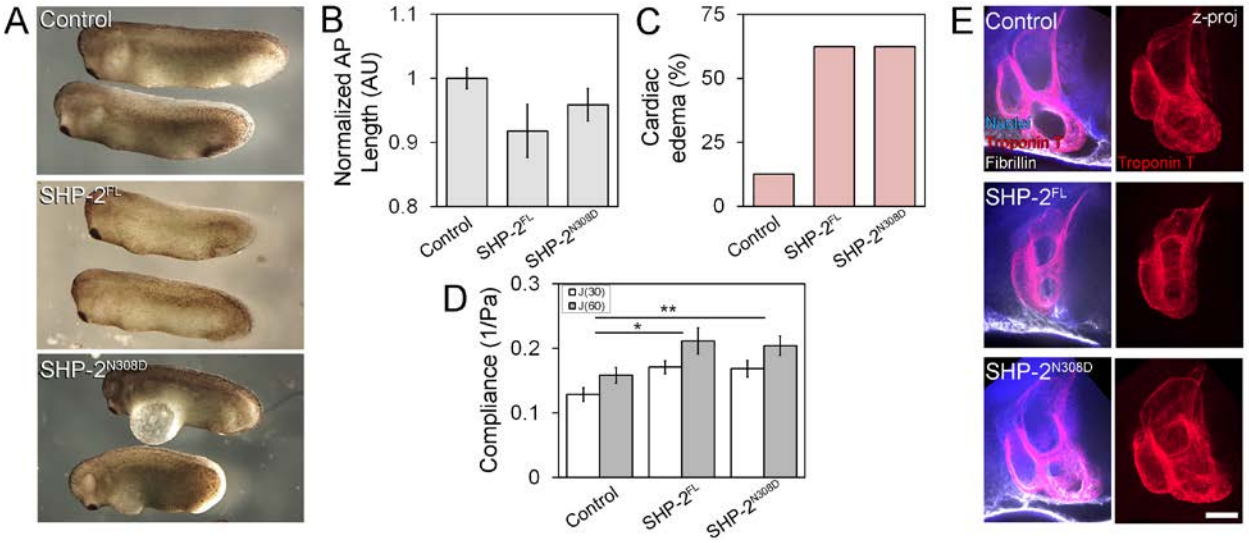

Figure 31. *In vivo* effects of SHP-2 overexpression

(A) SHP-2 expressing Stage 28 embryos show cardiac edemas (arrows). (B) SHP-2 overexpression reduces AP length and (C) increases number of cardiac edemas per clutch. (D) Both SHP-2FL and SHP-2N308D overexpression cause a 30% increase in tissue compliance at Stage 22, just prior to MET. (E) SHP-2 overexpression causes cardiac defects.

To determine whether SHP-2^{FL} and SHP-2^{N308D} related defects were cell-autonomous or microenvironmental, we targeted expression to either the endoderm or HPCs and fixed at Stage 37 to assess heart morphology based on size and shape, as done previously (Jackson, Kim et al. In review). Interestingly, we observe trending cardiac defects in all cases (Fig. 32). Defects associated with expressing SHP-2^{FL} and SHP-2^{N308D} in the endoderm are consistent with the idea that purely mechanical cues from the microenvironment can drive cardiac defects, although SHP-2 also appears to have autonomous effects on HPCs. Additional studies will need to be performed to see if this is consistent across multiple clutches.

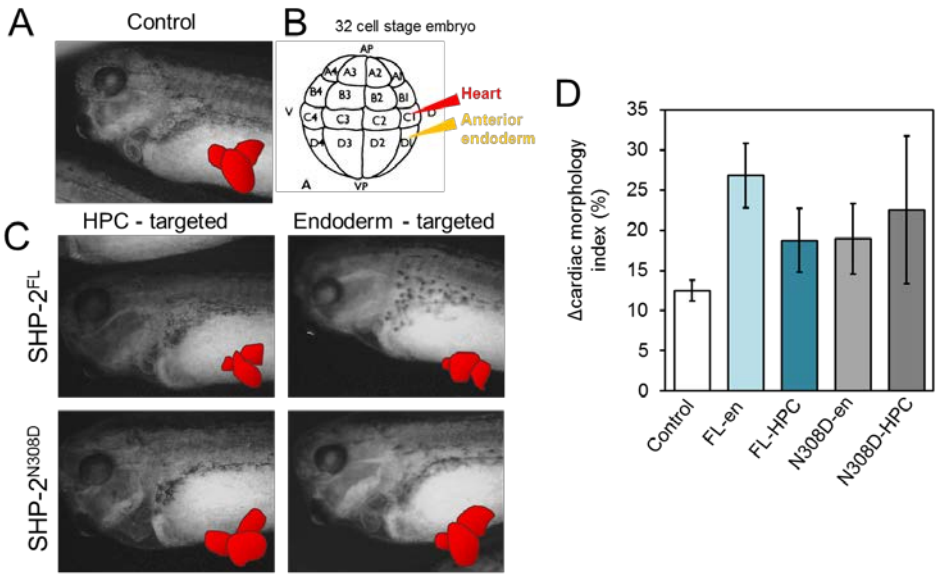

Figure 32. SHP-2 overexpression causes cardiac defects autonomously and non-autonomously

(A) Stage 37 control embryo with shape of heart shown in red. (B) Schematic of targeted injections. (C) Both endoderm-targeted and HPC-targeted expression of wildtype and mutant SHP-2 cause cardiac defects. (D) Semi-quantitative methods to assess cardiac morphology show high morphology variability due to SHP-2 overexpression.

5.2.2 SHP-2^{N308D} causes cell-autonomous abnormal MET

Because SHP-2 overexpression reduces bulk tissue compliance in the HFR and we have previously noted that a 30% increase in tissue compliance induced by MBS^{T695A} expression in the ventral anterior endoderm was sufficient to inhibit MET in HPCs (Jackson, Kim et al. In review), we wondered if mutant or wildtype SHP-2 overexpression would affect MET in HPCs. We fixed and sectioned embryos at Stage 28 at the onset of tropomyosin expression, which enables identification of cardiomyocytes (Kolker, Tajchman et al. 2000). Staining for epithelial markers aPKC and ZO-1, we then quantified apical epithelial marker intensity as done

previously (Jackson, Kim et al. In review). Consistent with our previous findings, SHP-2^{N308D} expressing embryos displayed an approximately 30% reduction in apical aPKC expression compared to the uninjected control (Fig. 33A-B). However, SHP-2^{N308D} expressing embryos had a nearly three-fold increase in ZO-1 expression (Fig. 33C-D), considerably higher than anything we have observed before (Fig. 20E). Interestingly, despite SHP-2^{FL} expression effects on tissue compliance, there was no difference noted in epithelial marker localization (Fig. 33A-D).

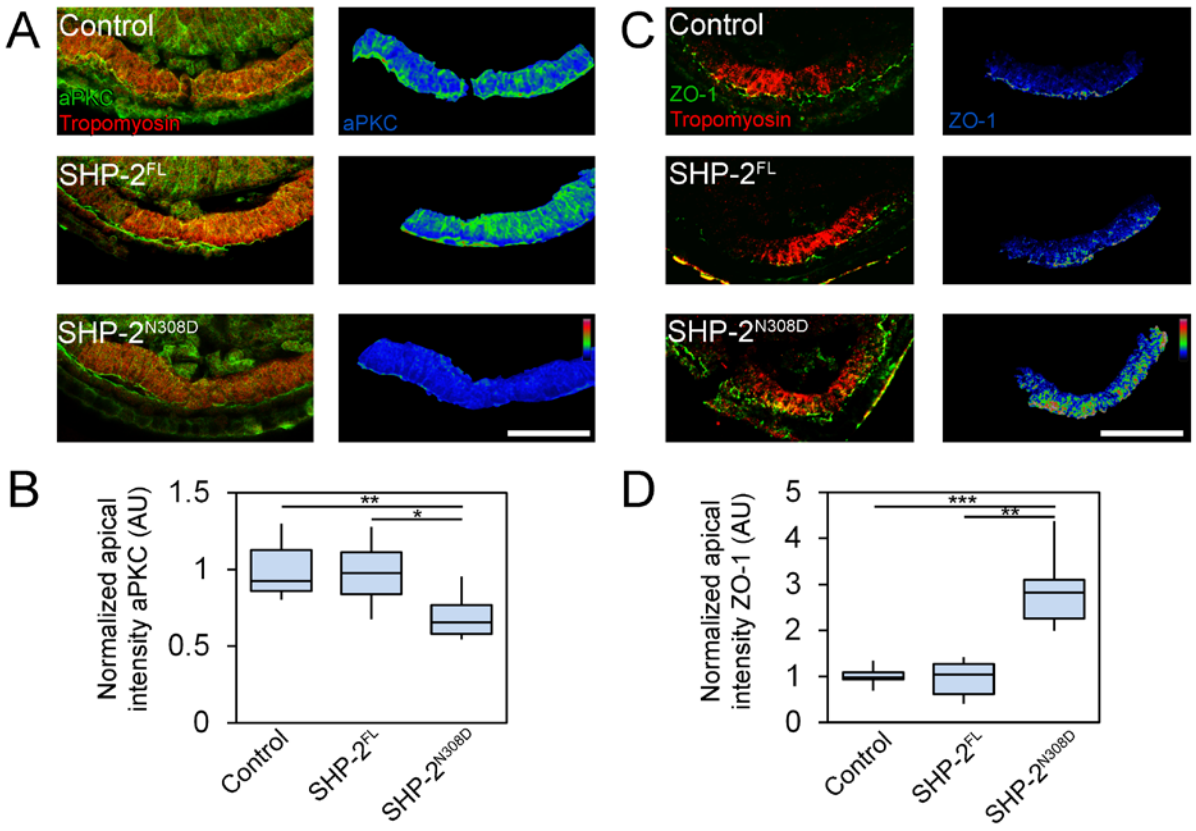

Figure 33. SHP-2^{N308D} overexpression causes abnormal MET

(A) aPKC (green) and tropomyosin (red) in tranverse sections of Stage 28 embryos. Pseudocolor LUT to emphasize aPKC intensity shown on right. (B) Measurement of normalized apical intensity of aPKC shows decrease in SHP-2^{N308D} expressing embryos. (C) ZO-1 (green) and tropomyosin (red) in tranverse sections of Stage 28 embryos. Pseudocolor LUT to emphasize ZO-1 intensity shown on right. (D) Measurement of normalized apical intensity of ZO-1 shows substantial increase in SHP-2^{N308D} expressing embryos.

To obtain better access to the molecular signaling pathways of MET, we turned to our embryonic mesenchymal cell aggregate model (Kim, Jackson et al. submitted) to determine if MET was similarly affected by SHP-2 expression. This 3D *ex vivo* model consists of mesenchymal cells from the deep ectoderm of the *Xenopus* gastrula, which spontaneously aggregate and rapidly undergo MET on the surface. Using apical actin structure to identify cells that have undergone MET on the surface of the aggregate as done previously (Fig. 34A), as apical actin demonstrates high correlation to ZO-1 localization (Kim, Jackson et al. submitted), we observe increased MET on the surface of the aggregate in SHP-2^{N308D} expressing cells (Fig. 34B). We also see a high correlation between cells expressing of SHP-2^{N308D} and having undergone MET that is not observed in the SHP-2^{FL} cells, suggesting SHP-2^{N308D} elicits a cell-autonomous cue to accelerate MET (Fig. 34C). Using qPCR to analyze expression of various epithelial genes, we find aPKC expression is undetectable in SHP-2^{N308D} aggregates, while ZO-1, ROCK1, ROCK2 and others are expressed at levels similar to the control (Fig. 34D). These results led us to suspect that SHP-2N308D causes a cell-autonomous abnormal MET, characterized by downregulated aPKC and increased tight junction assembly.

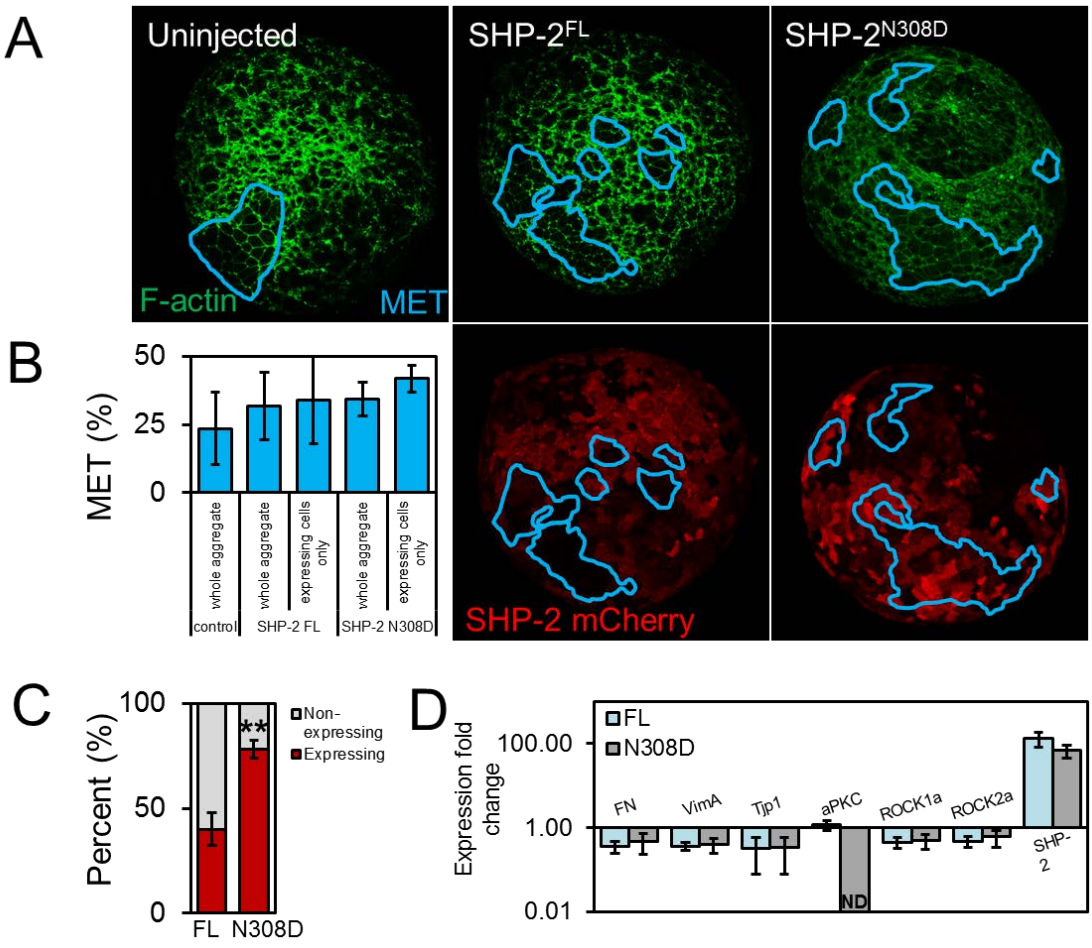

Figure 34. SHP-2^{N308D} overexpression causes cell-autonomous MET in mesenchymal cell aggregates

(A) 5hpa aggregates were fixed and stained for F-actin (green) and SHP-2 mCherry (red) to quantify the amount of MET on the surface of the aggregate (blue outlines). (B) Higher MET is seen in SHP-2^{N308D} expressing cells (N = 5-7 aggregates per group over one clutch). (C) Analyzing only cells that have undergone MET, there is a high correlation of MET and SHP-2^{N308D} expression not seen in SHP-2^{FL} (D) qPCR analysis of mesenchymal (FN and VimA), epithelial (Tjp1, aPKC), and ROCKs show most gene expression is unchanged, with the exception of aPKC in SHP-2^{N308D}, which is not detectable (ND).

5.2.3 Fibrillar fibronectin-β1 integrin signaling is necessary for HPCs to undergo MET

Because SHP-2^{N308D} cells appear to have a cell autonomous increase in MET, opposite of what is predicted by the measured increase in tissue compliance, we hypothesized that SHP-2^{N308D} may be perturbing the native mechanosensory pathways of HPCs. We have previously implicated the underlying endoderm as the source for endogenous cues driving MET (Jackson, Kim et al. In review), which implies the mechanosensor is likely at the HPC-endoderm interface. During HPC migration, this interface is rich with fibronectin which is necessary for HPC polarization (Yelon and Stainier 2005, Garavito-Aguilar, Riley et al. 2010). It was previously shown in *Xenopus* that injecting a 70kD mutant fibronectin construct could knock down the formation of fibronectin fibrils without interrupting the molecular binding domain of β1-integrin, which was sufficient to induce cardiac edemas (Rozario, Dzamba et al. 2009). Taken together, this generated our hypothesis that fibronectin is necessary for the transmission of mechanical information between the endoderm and HPCs. We first confirmed the involvement of β1-integrin and fibronectin in HPC MET by injecting the function-blocking antibody IF7, which prevents interaction of fibronectin and β1-integrin, into the HFR in Stage 22 embryos. HPCs in IF7 injected embryos showed highly perturbed polarity, exhibiting rounded cell shapes and no localization of aPKC or ZO-1 (Fig. 35).

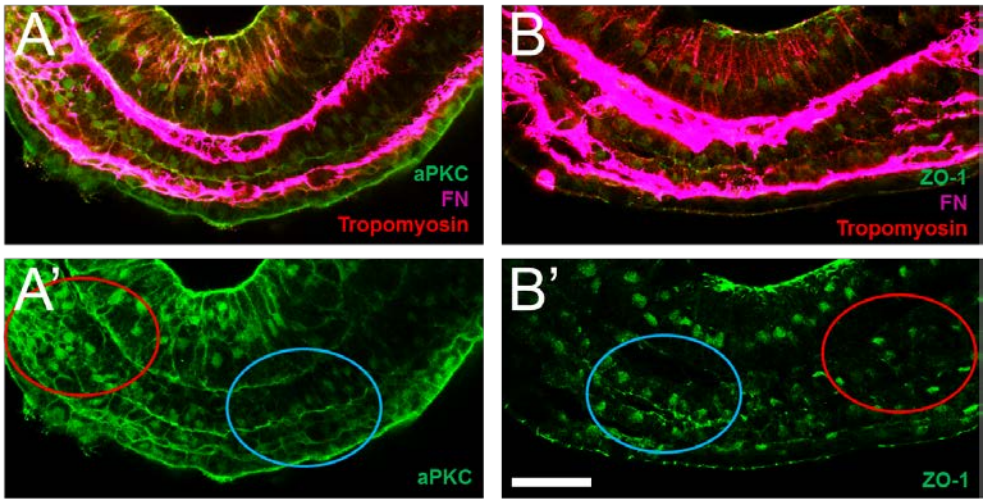

Figure 35. Blocking β1-integrin-fibronectin signaling with IF7 inhibits MET.

(A) Embryos were injected with the fibronectin function blocking antibody IF7 at Stage 22 into the HFR, fixed at Stage 28 and stained for FN (magenta), tropomyosin (red) and epithelial markers (green), either (A) aPKC or (B) ZO-1. (A'-B') Epithelial marker channel where red ellipse shows area of IF7 injection whereas blue circle shows region without perturbation. Scale bar = 50μm.

We suspected that fibrillar fibronectin acts to transmit mechanical forces between the endoderm and the HPCs through its interaction with β1-integrin. To further probe the mechanical role of fibrillar fibronectin, we injected the 70kD mutant mRNA into one blastomere of the 2-cell stage, to express the construct in half of the embryo. As a proof of concept, to determine if we correctly decoupled mechanotransduction between the endoderm and HPCs, we also targeted the constitutively-active arhgef2^{C55R} expression to the endoderm, which decreases tissue compliance in the HFR by approximately 40% (Jackson, Kim et al. In review). We note whole embryo defects (Fig. 36A), including a reduced AP length in all treatments and high incidence rates of cardiac edemas per clutch in the arhgef2^{C55R} expressing embryos (Fig. 36B), consistent with our previous report. In addition to measuring the normalized apical intensity of ZO-1 and aPKC, we also calculated the ratio of apical intensity between the two heart fields with the expectation that

we would observe a larger ratio in response to knocking down fibrillar fibronectin in half of the embryo (Fig. 36C). Consistent with our previous observations, arhgef2^{C55R} embryos show approximately 50% higher aPKC (Fig. 36D-E) and ZO-1 (Fig. 36F-G) apical intensity compared to the control. Both controls and arhgef2^{C55R} expressing embryos had field intensity ratios close to 1 (Fig. 36E and G). In 70kD injected embryos, we a significant 30% decrease in mean aPKC apical intensity and a slight increase in the field aPKC intensity ratio (Fig. 36E) while we see a slight 15% decrease in mean ZO-1 apical intensity and a significant increase in the field ZO-1 intensity ratio (Fig. 36G). Interestingly, the arhgef2^{C55R}+70kD embryos did not exhibit large differences in mean epithelial marker intensity compared to the control (Fig. 36E, G), presumably because averaging together the fibronectin knockdown field and the arhgef2^{C55R} stimulated field together canceled each other out. However, arhgef2^{C55R}+70kD mean apical intensities display a significant 50% decrease compared to the arhgef2^{C55R} only embryos (Fig. 36E, G), showing that the knockdown of fibrillar fibronectin effectively stopped the transmission of mechanical information from the stiffer arhgef2^{C55R} expressing endoderm to the HPCs. Furthermore, the field ratio of arhgef2^{C55R}+70kD embryos was significantly higher than all other treatments (Fig. 36E, G), illustrating the large epithelial marker expression discrepancy between the fibronectin knockdown and the arhgef2^{C55R} accelerated MET. Thus, we interpret these results as confirmation that fibrillar fibronectin is required to transmit mechanical information between the endoderm and HPCs.

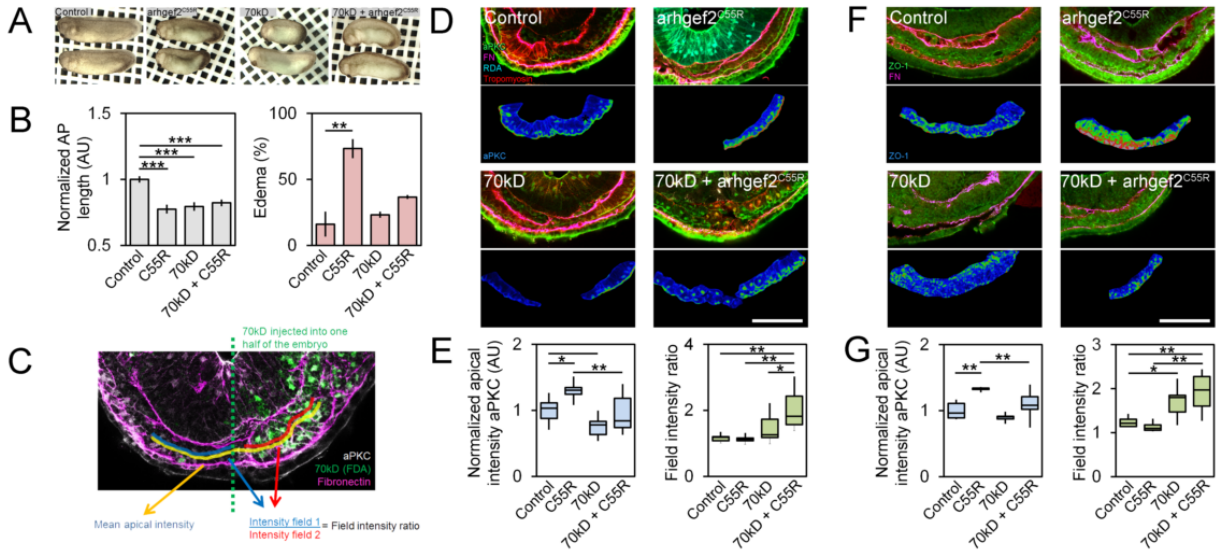

Figure 36. Knockdown of fibrillar fibronectin inhibits MET in HPCs.

(A) Embryos were injected with 70kD fibronectin mRNA to isolate expression to one half of the embryo and arhgef^{C55R} to target expression in the anterior endoderm. Whole embryos show (B) reduced AP length due to all treatments and increased incidence of cardiac edemas due to arhgef2^{C55R}. (C) Schematic for image analysis process (D) Top panel shows transverse sections of Stage 28 embryos stained for and fixed at Stage 28 and stained for FN (magenta), tropomyosin (red) and aPKC (green). Lower panel shows pseudocolor LUT to emphasize aPKC intensity. (E) Left panel shows mean normalized apical intensity measurements of aPKC shows aPKC increases due to arhgef2^{C55R} and decreases due to 70kD. Right panel shows the aPKC apical intensity ratio between the two fields, showing that expressing 70kD in half of the embryo is sufficient to inhibit MET in that half of the embryo, driving high ratio values, which cannot be rescued by arhgef^{C55R}. (F) Top panel shows transverse sections of Stage 28 embryos stained for and fixed at Stage 28 and stained for FN (magenta) and ZO-1 (green). Lower panel shows pseudocolor LUT to emphasize ZO-1 intensity. (G) Left panel shows mean normalized apical intensity measurements of ZO-1 shows ZO-1 increases due to arhgef2^{C55R}, which is reversed due to expression of 70kD. Right panel shows the ZO-1 apical intensity ratio between the two fields, showing that expressing 70kD in half of the embryo is sufficient to inhibit MET in that half of the embryo, driving higher ratio values, which cannot be rescued by arhgef^{C55R}. All scale bars = 50μm.

5.2.4 SHP-2^{N308D} overrides fibrillar fibronectin knockdown to upregulate tight junction formation

Now that we demonstrated an ability to inhibit transmission of mechanical cues between the endoderm and HPCs, we wanted to test our hypothesis that SHP-2^{N308D} results in a mechanosensory defect, causing cells to think they are always experiencing stiff environments. To perform this experiment, we targeted expression of SHP-2^{N308D} to HPCs and the 70kD construct to half of the embryo. Fixing and staining stage 28 embryos, we find preliminary evidence that SHP-2^{N308D} causes cell autonomous downregulation of aPKC and cell-autonomous increases in ZO-1 (arrows, Fig. 37). Furthermore, ZO-1 expression did not appear limited to the apical surface, suggesting that regulation of apicobasal polarity may be unstable. When fibrillar fibronectin is knocked down with the 70kD construct, we observe aPKC expression is not noticeably different in SHP-2^{N308D} expressing cells compared to non-expressing cells (Fig. 37). However, ZO-1 appears to be highly expressed due to SHP-2^{N308D} expression, although there appears to be no clear apical boundary so it is not clear if the ZO-1 expression reflects functional tight junctions or immature junctions. Thus, we have promising preliminary data suggesting SHP-2^{N308D} upregulates abnormal tight junction assembly regardless of the mechanical information received by HPCs. Further studies will establish a correlation between SHP-2^{N308D} expression, aPKC downregulation and ZO-1 upregulation in HPCs.

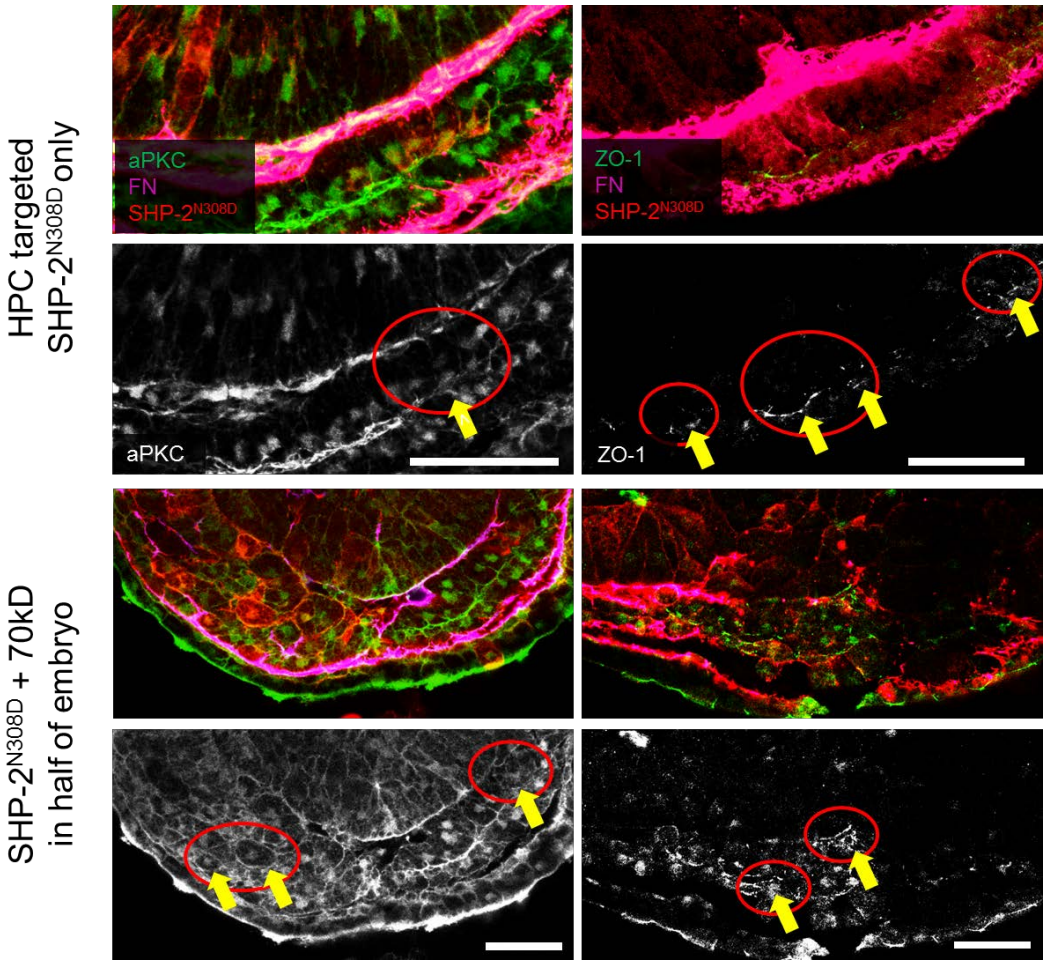

Figure 37. SHP-2N308D expression causes cell-autonomous abnormal MET in spite of fibrillar fibronectin knockdown.

Top panels show transverse sections of Stage 28 embryos stained for epithelial markers (green), either aPKC (left) or ZO-1 (right), fibronectin (magenta) and SHP-2^{N308D}-mCherry (red). Bottom panels show grayscale images of epithelial marker expression, with red ellipses indicating areas with SHP-2^{N308D} expressing cells and arrows indicate the epithelial marker expression corresponding to SHP-2^{N308D} expression.

5.3 DISCUSSION

Here, we report that SHP-2^{N308D} related CHDs may arise from errors in HPC MET, in stages prior to HPCs merging on the ventral midline. We demonstrate that wildtype and mutant SHP-2 expression is sufficient to increase tissue compliance in the heart forming region (Fig. 31), but only SHP-2^{N308D} causes errors in the timing of MET (Fig. 33). Furthermore, this MET is atypical, characterized by decreased aPKC activity, related to a combined downregulation of aPKC transcription (Fig. 34), localization (Fig. 33, 36) and phosphorylation , but upregulated tight junction assembly (Fig. 33, 36). Although expression of SHP-2^{N308D} in the endoderm is sufficient to drive cardiac defects (Fig. 32), the errors in MET may be cell autonomous (Fig. 34, 36). To confirm this, we identified the fibrillar fibronectin as the agent transmitting mechanical information between the endoderm and HPCs and found that SHP-2^{N308D} expressing HPCs upregulate tight junction formation in spite of fibrillar fibronectin knockdown. Previous studies have related errors in HPC proliferation to SHP-2^{N308D} (Langdon, Tandon et al. 2012). It is possible then that decreased proliferation is a direct consequence of the abnormal MET induced by SHP-2^{N308D}, although this relationship will require further testing to confirm.

It was unexpected to observe cardiac defects in response to both wildtype and mutant SHP-2 overexpression, although a similar phenomenon was reported in MDCK cells (Tien, Lee et al. 2016). Here, both SHP-2^{FL} overexpression and expression of the oncogenic mutation SHP-2^{E76G} caused errors in epithelial lumenogenesis, with SHP-2^{E76G} causing multiple sites of abnormal apical-basal polarity. They note that attenuation to aPKC activity, through inhibited phosphorylation of Par3, led to increased desphosphorylation of Tuba to reduce Cdc42 activity and increase ROCK2 phosphorylation (Tien, Lee et al. 2016). It was also observed that a similar constitutively active mutation, SHP-2^{E76R}, decreased FAK phosphorylation and causes disrupted

actomyosin dynamics and mislocalization of ZO-1 in Sertoli cells in rat testis (Puri and Walker 2013). Thus, while both wildtype and mutant SHP-2 causes defects, only mutant SHP-2 causes abnormal polarization.

As a ubiquitously expresses phosphatase, SHP-2 plays a central role in numerous signaling pathways, including PI3K, Ras and Sprouty, STAT3, AKT and mTOR and FGF. However, the sensitivity of HPC MET to SHP-2^{N308D} and other NS-related mutations eliminates several of these pathways, including AKT, FGF and PI3K. SHP-2^{N308D} expression does not alter phosphorylation levels of AKT or levels of GAB1 and p85, a docking protein involved in PI3K and the regulator unit of PI3K, respectively (Edouard, Combier et al. 2010). FGF signaling does not appear important during these stages of early heart development, as we have seen no evidence of MET or cardiac defects resulting from treatment with the FGF inhibitor SU5402 between Stages 20 and 28, during HPC movement to the ventral midline (Fig. 37). STAT3 phosphorylation does appear to be downregulated due to SHP-2^{N308D} (Zhang, Chan et al. 2009) and to our knowledge, neither activity or expression levels of Ras and Sprouty have been investigated. By contrast, the observation that ROCK inhibition rescues CHDs in SHP-2^{N308D} embryos (Langdon, Tandon et al. 2012) and that SHP-2^{N308D} perturbs apicobasal polarity establishment in HPCs (Fig. 33, 36) suggest that these are more likely the source of the defect. Further investigations will be necessary to rule out Ras-Sprouty signaling as a contributor to these defects.

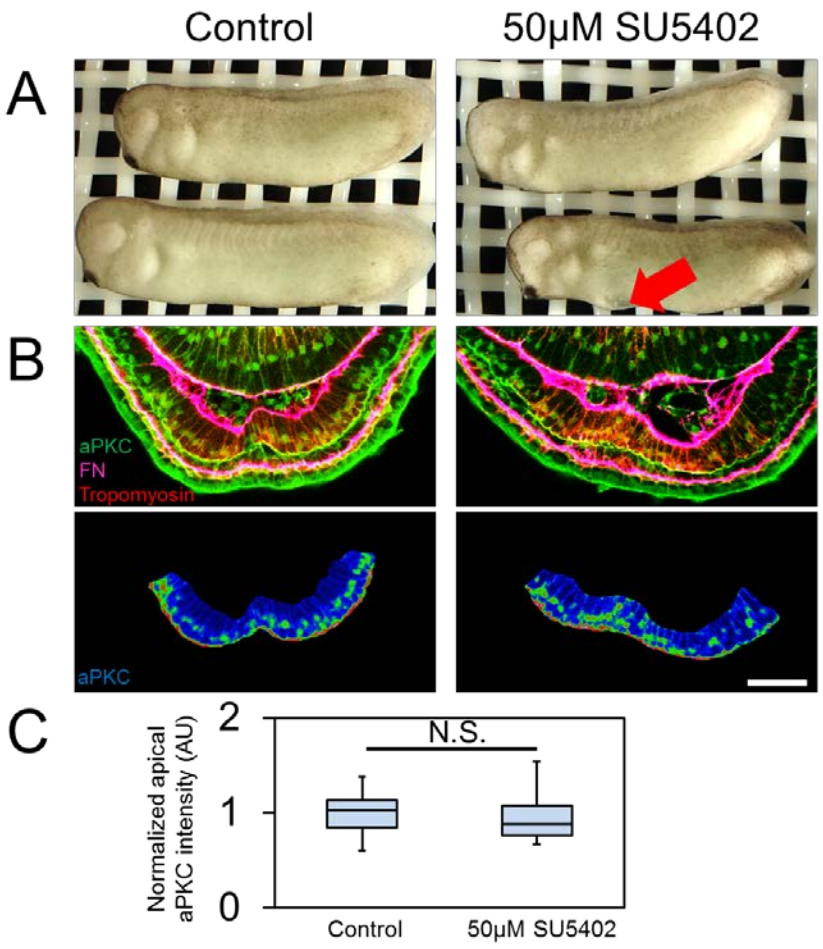

Figure 38. Inhibiting FGF signaling during early heart formation does not perturb MET.

(A) Stage 28 embryos treated with DMSO (control) or FGFR inhibitor SU5402 from Stages 20-28. Most SU5402-treated embryos look normal, with some rare instances of ventral edemas (arrow). (B) Top panel shows transverse sections of Stage 28 embryo stained for aPKC (green), fibronectin (magenta), and tropomyosin (red). Bottom panel shows cardiomyocyte-masked aPKC in pseudocolor LUT. Scale bar = 50um. (C) Quantification of apical aPKC shows no difference between control and SU5402 treated embryos (N = 6 each, over 2 clutches).

Par3 is a promising candidate for mediating SHP-2^{N308D} hyperactivation of tight junction assembly while simultaneously decreasing aPKC. Par3 localization within the cell is a critical early step of tight junction assembly, remaining at tight junctions after dissociating with Par6 and aPKC to demarcate apical vs. lateral domains (Zihni, Mills et al. 2016). Par3 also promotes YAP dephosphorylation to drive YAP translocation to the nucleus (Zhang, Wang et al. 2016), a known effector of mechanotransduction (Dupont, Morsut et al. 2011, Morikawa, Zhang et al. 2015) that can stimulate cytoskeletal remodeling (Bai, Zhu et al. 2016). Furthermore, Par3 interacts with the Rac-GTPase Tiam1, which in turn associates with integrins and talin at focal adhesion sites (Chen and Macara 2005). Depletion of Par3 in MDCK cells results in increased Tiam1 activity to hyperactivate Rac and cause aberrant actin cytoskeleton organization during tight junction assembly (Chen and Macara 2005). In its role as a GTPase exchange factor for Rac, Tiam1 is has also been implicated in mechanotransduction. For example, mechanical strain in lung cells induces Tiam1 translocation from the membrane to the cytoplasm to lower cell migration (Desai, Chapman et al. 2008). Furthermore, the involvement of cell-ECM adhesions as β1-integrin induces Par3 signaling during MET in endothelial cells (Zovein, Luque et al. 2010) suggests a mechanotransduction pathway whereby forces exerted on integrins regulate SHP-2, to balancing Par3 and opposing Tiam1 activity, which leads to apical polarization and tight junction assembly. By this hypothesis, SHP-2^{N308D} perturbs a cell's interpretation of mechanical forces, prohibiting Par3 and aPKC localization while allowing for unpolarized, unregulated tight junction assembly. Future mechanistic studies will be needed to test this hypothesized pathway (Fig. 39).

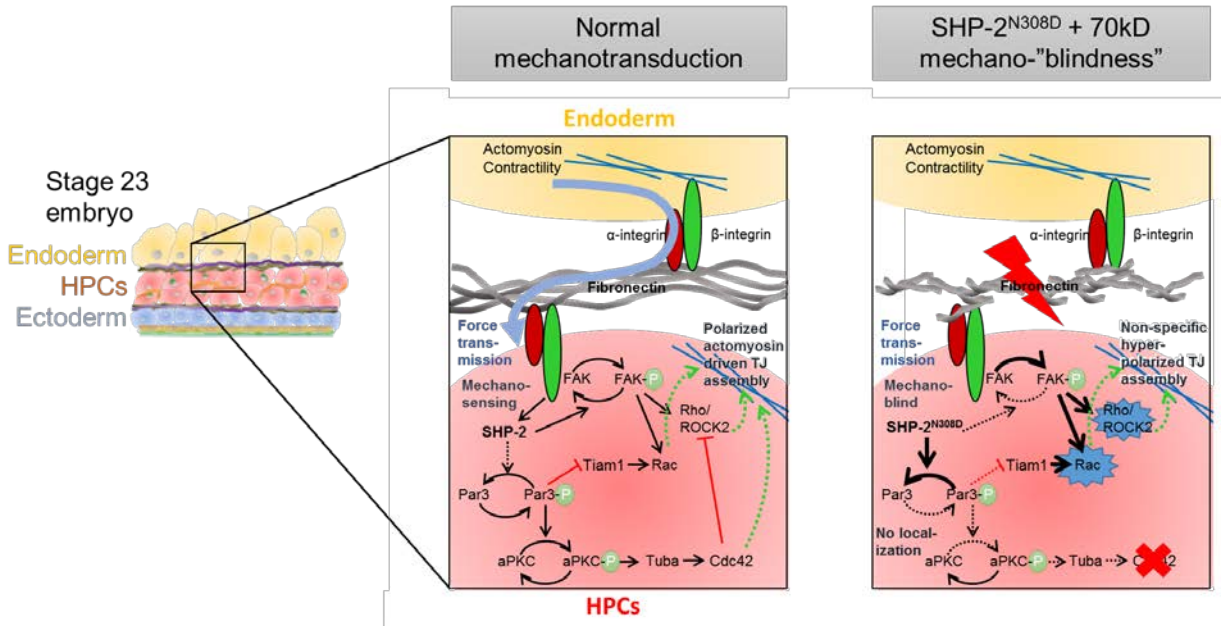

Figure 39. Hypothesized mechanism of SHP-2 signaling in mechanotransduction and HPC MET.

5.4 EXPERIMENTAL PROCEDURES

5.4.1 *Xenopus* embryos and microinjections

Xenopus laevis embryos were obtained via standard methods (Kay and Peng 1991) and cultured in 1/3× modified Barth Solution (MBS; (Sive, Grainger et al. 2000)). Microinjections we carried out in MBS supplemented with 3% Ficoll at either at the 1-cell (~1 ng SHP-2^{FL} and SHP-2^{N308D}), 2-cell (~270 ng 70kD fibronectin) or 32-cell stage (~30 pg arhgef2^{C55R}, ~80 pg SHP-2^{FL} and SHP-2^{N308D}). Targeted injections at the 32-cell stage were performed in accordance with *Xenopus* fate mapping experiments (Dale and Slack 1987, Bauer, Huang et al. 1994). Accuracy of targeted injections was verified twice, between Stages 20-23 using a fluorescence-

equipped stereomicroscope (Olympus) and again post-immunofluorescence staining using a confocal microscope (Leica SP5). Images of *Xenopus* embryos were taken with a CCD camera (QImaging) mounted on a stereomicroscope.

5.4.2 Immunofluorescence and microscopy

Embryos were devitellinized and fixed with 4% paraformaldehyde for visualizing aPKC and ice cold Dent's fix (4:1 of Methanol: DMSO) for visualizing ZO-1. Fixed aggregates were washed for 30 minutes with PBST and blocked with 10% goat serum in PBST for 1 hour prior to antibody staining. Primary antibodies for FN (4H2), aPKC (nPKCζ (C-20) sc-216; Santa Cruz), ZO-1 (Invitrogen) and fibrillin (JB3; Developmental Studies Hybridoma Bank) were used with 1:200 dilution. Antibodies for mCherry (3A11) and tropomyosin (CH1) were used at a 1:50 dilution while troponin T (CT3) was used at a 1:100 dilution (Developmental Studies Hybridoma Bank). Antibodies were then incubated overnight on a nutator at 4°C. After washing, the samples were incubated with the appropriate secondary antibody at a 1:200 dilution overnight at 4° C. Cell nuclei were visualized with Hoechst 33342 (1:2000, Thermo Fisher).

For aggregate MET studies, aggregates were fixed with 4% paraformaldehyde and 0.25% glutaraldehyde in PBST (1x PBS with 0.1% Triton X-100) for 15 minutes at room temperature. After washing, the samples were incubated with BODIPY-FL–phallacidin (1:800) to visualize F-actin. Cells were evaluated as having undergone MET if they exhibited no F-actin protrusions and had strong F-actin localization to cell boundaries, which is indicative of co-localization with ZO-1. Cells were determined to be SHP-2 positive if mCherry intensity was above a threshold kept consistent throughout one experimental clutch.

Fixed immunofluorescence imaging was performed using a laser-scanhead mounted on an inverted compound microscope (Leica SP5). Images were then imported into ImageJ (Schneider, Rasband et al. 2012) for image analysis, including quantification of apical intensity, and cardiac anatomy. In instances where fibronectin was co-stained with tropomyosin or mCherry, all of which are mouse IgG1 antibodies, fibrils were manually thresholded out of the channel and subjected to a different LUT. Quantification of apical intensity of epithelial markers has been previously described (Section 3.4.3; Fig. 9). In brief, cardiomyocytes were manually segmented out using tropomyosin expression or anatomy. Normalizing the epithelial marker intensity to background intensity in endoderm cells and then applied a 2-pixel wide Gaussian blur, intensity was measured in a 10-pixel wide linear region manually drawn to span the apical surface of the cardiomyocyte field. When calculating the ratio between fields, apical intensity of the left and right fields was measured separately.

5.4.3 SHP-2 mRNA

Xenopus laevis SHP-2 (Ptpn11) was amplified by PCR from a whole embryo Stage 10 cDNA library using primers Ptpn11-F (CAC CTG TCG CCA TCG ATA TGA CAT CCA GGA GGT GGT TTC) and Ptpn11-R (CTT GAA TTC GAA TCG ATT CAT CTG AAG CTT TTC TGC TGT TG). The ~2 kb DNA was gel-excised (Zymo Research) and subcloned by InFusion cloning (Clontech) into pCS2+ mCherry-LL2 digested with ClaI. Upon successful integration, the Ptpn11 sequence is fused in frame to the 3’ end (C-terminus) of mCherry.

To change the amino acid sequence to mimic the human N308D mutation, the nucleotide A at position 856 was mutated to G using an overlapping primer PCR strategy and the primers

xlPtpn11 N308D_F (CAT CAA TGC CGA CAT TAT TAT GCC TGA ATT TG) and xlPtp11 N308D_R (GCA TAA TAA TGT CGG CAT TGA TGT AAT CAG AG). All mutations were sequence verified.

To make RNA, 3 µg of DNA of each construct (pCS2+ mCherry-ptp11.S and pCS2+ mCherry-ptp11.S N308D) was digested with NotI, digested DNA purified (Zymo Research), and used as template in an in vitro RNA transcription reaction with Sp6 (AmpliCap Sp6 High Yield Message Maker Kit, CellScript). Transcription reactions were cleaned up (Zymo Research), RNA concentration determined, and analyzed for possible RNA degradation before injection.

5.4.4 Compliance measurements using micro-aspiration

Microaspiration was performed using a custom dual-reservoir microaspirator apparatus previous described (Sections 3.4.4 and 4.4.4) and illustrated in Fig. 20. In brief, a suction pressure was applied to the ventral HFR of the embryo as displacement into a microchannel was tracked. From the displacement data, a power law model for creep compliance implemented in MATLAB (Mathworks) was used to obtain time-dependent compliance measurements for each embryo. Material within a 150 uM depth contributes to the measurement (von Dassow, Strother et al. 2010), so the measured compliance reflects a composite mechanical property of all three germ layers within the HFR.

5.4.5 qPCR

qPCR protocols were described previously (Section 4.4.5). In brief, RNA was isolated from 10 aggregates in each group (control, SHP-2FL and SHP-2N308D) and cDNA was obtained using reverse transcription. qPCR was carried out according to the SsoAdvanced Universal SYBR Green Supermix (Bio-Rad) manufacturer's instructions, using the Bio-Rad iQ5 thermal cycler and PCR detection system using *Xenopus laevis* primers for GAPDH, H4A, Tjp1, VimA and FN (Table 1) and aPKC, Rpl8, ROCK1, ROCK2 and SHP-2 (Table 2). Fold changes due to SHP-2 overexpression were calculated using the Livak method, using H4, GAPDH and Rpl8 as reference genes and the control aggregate as a calibrator (Šindelka, Ferjentsik et al. 2006).

Table 2. qPCR Primers for SHP-2 experiments

Gene	Alternate Name	XenBase Name	Accession	Forward Primer	Reverse Primer
Rpl8		rpl8.S	XB-GENE-482213	TCCGTGGTGTGGCTATGAATCC	GACGACCAGTACGACGAGCAG
ROCK1*		Rock1.S	XB-GENE-6488328	AATTGAAGGCTGGCTCTCTATAC	GGCTCTGTATACATCTCCTTGTG
ROCK2&		Rock2.L	XB-GENE-959454	AGATGGCGCTAGATAGCAAAG	CCTGCTGCTTACCACTACATAC
aPKC	Prkci	prkci.L	XB-GENE-964810	GTCACCCTCAGACAGGATTTG	CTGACTGGTCAATCTTCCTCAC
SHP2	Ptpn11	ptpn11.S	XB-GENE-6254179	AGATCTCAGAGGTCAGGAATGG	GGAAGCGTAGGTGTTGGTAATG

* Only 1 nt mismatch in reverse primer compared to Rock1.L RNA, primer pair will likely amplify Rock1.L (XB-GENE-17336014) RNA as well.

& Only 1 nt mismatch in reverse primer compared to Rock2.S RNA, primer pair will likely amplify Rock2.S (XB-GENE-17340948) RNA as well.

6.0 SUMMARY AND CONCLUSIONS

6.1 SUMMARY OF FINDINGS

We proposed three specific aims in order to eludicate the role of mechanical signaling in regulating normal heart development and CHDs. Early in this project, we characterized the progression of MET in HPCs, which initiates at Stage 23 or approximately halfway through their movement from the anterior lateral plate mesoderm to the ventral HFR (Section 3.2.1). Recognizing HPC MET as a crucial event in early heart development, we examined how mechanical cues drive this process. Chapter 2 focuses on reviewing the role of mechanics in driving METs in several cases in development, to provide context for Aim 2, including early heart development (Section 2.5.2), kidney development (Section 2.5.1) and somitogenesis (Section 2.5.3), as well as in diseases such as cancer (Section 2.9). We noted trends in the MET process and propose shared mechanisms across cases of MET (Section 2.10).

In Chapter 3, we directed our attention to directly test hypotheses developed within Aims 1 and 2. We address Aim 1, the question of how HPC and ventral tissue movements relate to MET and contribute to overall heart development, in sections 3.2.2-3.2.4. Using nuclei tracking we find that HPC migration trajectories begin this process as relatively fast, persistent and collective before transitioning to slower and more autonomous (Fig. 13B-E). These behaviors may in part be explained by mechanical coupling to the endoderm, as shown by our low

resolution traction force microscopy experiments (Fig. 13F-H). In addition to tracking HPC movements, we also tracked deformations of the entire ventral HFR using to show that strain peaks at Stage 23 (Section 3.2.4). Then using microaspiration to measure compliance, we find that compliance drops and consequentially, environmental stress peaks, at Stage 23 (Fig. 14).

Turning to Aim 2, we first perform experiments to alter bulk tissue mechanics and show HPC MET is directly correlated to bulk tissue mechanics (Section 3.2.2). Furthermore, we find that inducing or inhibiting contractility in the endoderm to modulate the tissue compliance sensed by HPCs, we can accelerate or delay MET (Section 3.2.5). We also demonstrate that exogenous stress, induced by injection of mineral oil into the archenteron, is sufficient to drive MET in HPCs (Section 3.2.6). In Chapter 4, we expand the application of Aim 2 to a different model, the 3D mesenchymal cell aggregate, and describe similar trends in how mechanics regulate MET.

Because we see MET perturbations giving rise to specific physiological defects (Section 3.2.7), we were encouraged to pursue the SHP-2^{N308D} mutation, known to give rise to a congenital heart defect associated with Noonan Syndrome, and investigate its effects on MET. Thus, we address Aim 3 in Chapter 5. We demonstrate an ability to recreate the Noonan Syndrome CHD phenotype in *Xenopus* embryos by overexpressing the SHP-2^{N308D} mRNA in embryos (Section 5.2.1). We also observe a 30% increase in tissue compliance in both wildtype and mutant SHP-2 expressing embryos (Fig. 31), which may in part explain the resulting CHDs. We also show that SHP-2^{N308D} does indeed perturb MET, although in a manner that was quite unexpected. We observe decreased aPKC but a “hyperpolarized” ZO-1 (Section 5.2.2). By first confirming the role of fibrillar fibronectin as the agent transmitting mechanical forces between the endoderm and HPCs (Section 5.2.3), we also show that SHP-2^{N308D} causes abnormal MET

independent of the native mechanosensory framework. We also propose novel integration of mechanic and conventional signaling pathways to explain these findings (Section 5.2.4), which will require more mechanistic studies to test and confirm.

6.2 SIGNIFICANCE OF FINDINGS

Early heart development, in spite of its crucial role of setting the foundation for the rest of heart organogenesis, remains poorly understood. One of the most significant results of this study was the establishment of our experimental model for early heart development (Fig. 20). Through application of this model, we place the MET of HPCs at a crucial juncture of HPC movements to the midline, demonstrating that temporal regulation of HFR mechanics is necessary to proper heart structure and function. By connecting the development of CHDs to altered timing of MET, both through purposeful modification of tissue compliance and through expression of a clinically-relevant genetically-mutated protein, we shed light on the importance this MET and errors in early heart development in general on the etiology of CHDs. Many CHDs are tied to later defects, including defects in cilia during periods of blood flow (Brueckner 2007) and defects in HPC proliferation post-heart tube stage (Langdon, Tandon et al. 2012). However, as our research suggests, these may be the consequences of earlier lesions in cell polarity.

Additionally, the process of MET is central to numerous developmental processes (Kim, Jackson et al. 2016). Shedding light on the role of mechanical cues in initiating MET in HPCs and mesenchymal cell aggregates suggests that a shared MET framework may exist. By demonstrating the crucial role of mechanics in MET in HPCs, we may motivate future work to

investigate similar studies in kidney development, cancer studies or many others. This will further the acceptance of mechanics as an integral part in driving morphogenesis, which will ultimately enable a more complete understanding of developmental processes and how birth defects may arise.

Studying early stages of heart development has been logistically hindered by the fact that post-gastrulation HPCs do not readily express any sort of marker until they nearly reach the ventral HFR. Furthermore, HPC movement to the ventral midline is mostly obscured because it occurs deep within tissues. Here, capitalizing on the robustness of *Xenopus* development, we demonstrate microsurgical methods which permit visualization of HPC movements using an intravital and *ex vivo* model. Establishment of these methods will hopefully enable future studies to probe even further into the mechanical and molecular signaling pathways driving HPC movements to the HFR.

6.3 FUTURE DIRECTIONS

Although this research has elucidated some aspects of early heart development and MET, there are a number of questions that have arisen from it as well, many of which I list below.

6.3.1 Do HPCs sense stiffness or tension?

During early heart development, the underlying endoderm is converging and extending, which is known to generate a stiffness gradient (Varner and Taber 2012) but also likely exerts

tension on the fibronectin ECM that could be sensed by HPCs. Is it possible to parse out whether HPCs require active forces or passive stiffness to drive MET? Not included in this dissertation were preliminary experiments where I attempted to put windowed embryos on polyacrylamide gels of varying stiffnesses. Unfortunately, technical challenges limited their utility. Windowed whole embryos bound to polyacrylamide did not yield consistent and reliable results. Furthermore, due to the presence of pericardial precursors adhered to the gel rather than the myocardial precursors, the interpretation of these experiments would be difficult. However, this question remains and I think it to be an important one. Our group is currently developing a technology to inject embryos with protein-conjugated beads that can be moved in response to a magnetic field. Perhaps this question could be pursued using this method, conjugating B1-integrin to the bead surface and targeting them to the HPC-endoderm interface. Non-moving beads could serve as a passive stiff substrate, while moving ones could exert force on HPCs.

6.3.2 Does MET induce the migratory behavior changes or vice versa?

One limitation of our current analysis into the relationship between HPC migratory behaviors and MET is that it is strictly correlative. Determining a causative relationship between these two phenomena would be very enlightening. However, our current methods for altering MET in these cells is limited to targeting actomyosin contractility, which also likely alters cell migration to some extent. To investigate this question, we may require a purely molecular way to perturb MET timing and see how that affects migration. Perhaps overexpression or uncaging expression of a mesenchymal gene, such as Snail, it may be possible to stop MET in HPCs and characterize how it affects migration. While our data suggests that MET directly influences

HPCs to move slower and detach from the endoderm, it would be enlightening to have stronger causative evidence to support that.

6.3.3 How universal is the "mechanics induces MET" framework in development?

Development is full of cases of MET (Kim, Jackson et al. 2016). It is very tempting to consider that each MET has mechanical elements initiating them, but that hypothesis likely needs to be tested. Somitogenesis involve paraxial mesoderm extending posteriorly (Palmeirim, Henrique et al. 1997), which likely exert tension on the cells before the undergo MET. Kidney tubule development occurs via a similar mechanism where the elongating tubule is led by mesenchymal cells and follower cells undergo MET (Atsuta and Takahashi 2015). Are there tension/stiffness changes involved in these processes and are they required for MET? If we discover that this framework is consistent across these different organogenesis programs, perhaps it will identify potential mechanisms to treat MET related birth defects and diseases.

6.3.4 How important are other focal adhesion proteins in HPC mechanosensing?

Literature points to the role of talin, vinculin, α-integrins and FAK as potential mechanosensors (Wang, Dembo et al. 2001, Friedland, Lee et al. 2009, Grashoff, Hoffman et al. 2010, le Duc, Shi et al. 2010, Bogatan, Cevik et al. 2015, Yan, Yao et al. 2015). Which ones are necessary for HPC MET and which ones are disposable? Are any/all of these focal adhesion proteins upregulated in response to endodermal stiffening? Again optogenetic tools may be ideal to knock down or upregulate these proteins with spatiotemporal specificity during HPC

movements to the ventral midline. There is still much to discover about the so-called “mechanosome” and it would be interesting to exactly how HPCs convert mechanical cues into molecular signals. Identifying key players may offer deeper insight into the etiology of CHDs.

APPENDIX A

EXTRACELLULAR ATP REGULATES CONTRACTILITY OF EARLY EMBRYONIC EPITHELIA

This appendix describes my contribution to a manuscript first started by Sagar Joshi that is currently being prepared for submission. Although cell signaling and mechanical forces play vital roles in morphogenesis and organogenesis, pathways that trigger the generation of forces shaping embryonic tissues are not well understood. We have previously utilized nano-perfusion assays to demonstrate that cell lysate can induce F-actin remodeling and epithelial contractions (Joshi, von Dassow et al. 2010, Kim, Hazar et al. 2014). Prior to my involvement in this study, ATP was identified as the active component in lysate, participating in a G-protein coupled receptor (GPCR) pathway to regulate the contractile tone of epithelium during gastrulation. Using nano-perfusion techniques on *Xenopus laevis* embryos, it was found that acute exposure produced spatiotemporally-specific epithelial contractions while chronic exposure caused exogastrulation. Of the possible receptors for extracellular ATP, it was demonstrated demonstrated that GPCRs of the P2Y family were likely candidates by expressing Gγ-SaaX mutant mRNA in embryos, of form of Gγ proteins that bind to Gβ proteins and trap them in the cytoplasm and preventing their localization and activation at the plasma membrane (Mulligan, Blaser et al. 2010). Expressing Gγ-SaaX in embryos reduced or abolished contractility. Temporal

expression of P2Y-family proteins, analyzed using PCR, indicated P2Y2R as the target receptor. Using various inhibitors of F-actin and non-muscle myosin II, we showed actomyosin contractility is a key target in the response to ATP.

Here, I demonstrate that there is endogenous extracellular ATP that regulates blastopore closure, which is released in response to mechanical stress. Furthermore, I confirm native expression of P2Y2R in the gastrula animal cap. Using antisense knockdown, I demonstrate that contractility is a direct result of P2Y2R signaling, which also controls bulk tissue mechanics of the gastrulating embryo, and finally confirm the accuracy of our morpholino with rescue experiments.

A.1 RESULTS

A.1.1 Endogenous extracellular ATP regulates gastrulation

After we observed epithelia contracting in response to extracellular ATP, we wondered if endogenous extracellular ATP had a role in gastrulation. Using the ATP hydrolyzing enzyme apyrase, filmed blastopore closure (Fig. 40A-D) after incubating embryos with 130U/mL apyrase starting at Stage 8. Using ImageJ software, we measured the blastopore area of embryos manually and found that embryos incubated with apyrase had an approximately 20% smaller blastopore after 7 and 9 hours (Fig. 40E), indicating that gastrulation was accelerated due to knockdown of extracellular ATP. Exposing embryos to ATPγS, a non-hydrolysable form of ATP, we confirmed that chronic exposure to extracellular ATP prevented the blastopore from closing. As expected, embryos treated with both ATPγS and apyrase also had a larger blastopore

area. Taken together, these results show that endogenous extracellular ATP is present during gastrulation, to regulate blastopore closure through epithelial contractility.

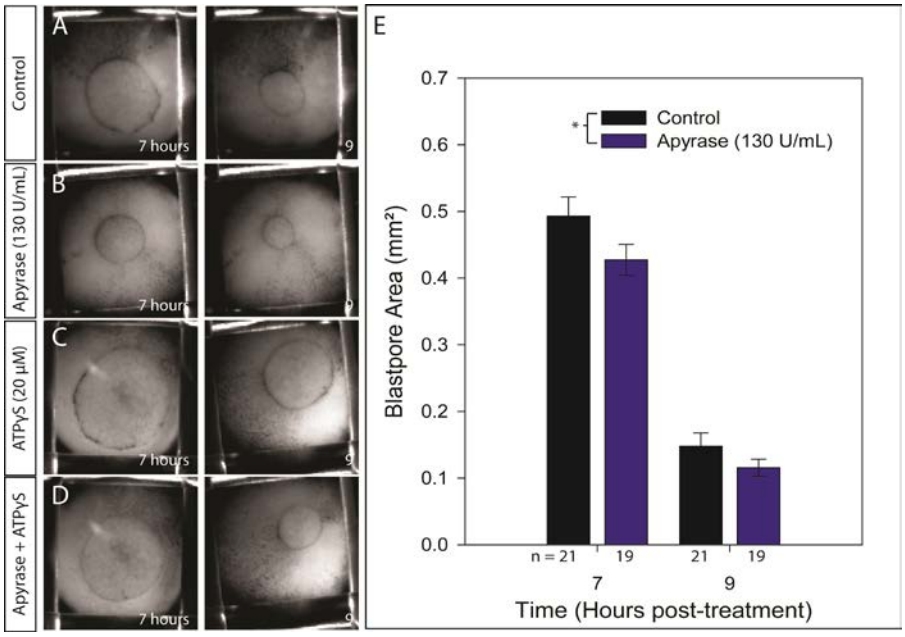

Figure 40. Endogenous extracellular ATP regulates blastopore closure.

Representative gastrulating embryos are shown treated with (A) control, (B) apyrase, (C) ATPγS, and (D) apyrase + ATPγS. Embryos treated with the apyrase, which hydrolyzes ATP, finished blastopore closure significantly faster than control embryos, as demonstrated by an one-way ANCOVA on the final measured area, using the initial area as a covariate, with $p < 0.05$.

A.1.2 Mechanical compression is sufficient to drive release of extracellular ATP

Because we had only observed epithelial contractions in response to wounding, we wondered if wounding was required for the release of extracellular ATP or if embryos would release ATP in response to mechanical stress. To investigate this, we devised a method of applying a gentle compression onto gastrulating embryos using agarose molded into a PCR tube (Fig. 41A). In order to measure the concentration of ATP in the media due to compression, we

inset a micropipette tip into the agarose mold (Fig. 41C), removed the tip to create a channel (Fig. 41D, white lines) and applied a compressive force taking care not to cause damage to the embryo (Fig. 41E). Then we used a commercially available ATP Bioluminescence Assay Kit to calculate the concentration of ATP in the media surrounding an embryo (Fig. 41B).

When compressed, we find that concentrations of ATP in the media surrounding the embryo were about 100nM, more than 10-fold higher than the sham (shorter agarose mold that would not compress the embryo) and 100-fold fold higher than the control (Fig. 41B). As expected, lysing embryos result in a very large concentration of ATP in the media, over 10,000-fold higher than the control at nearly 10μM. These results suggest that embryos release ATP in response to a mechanical force.

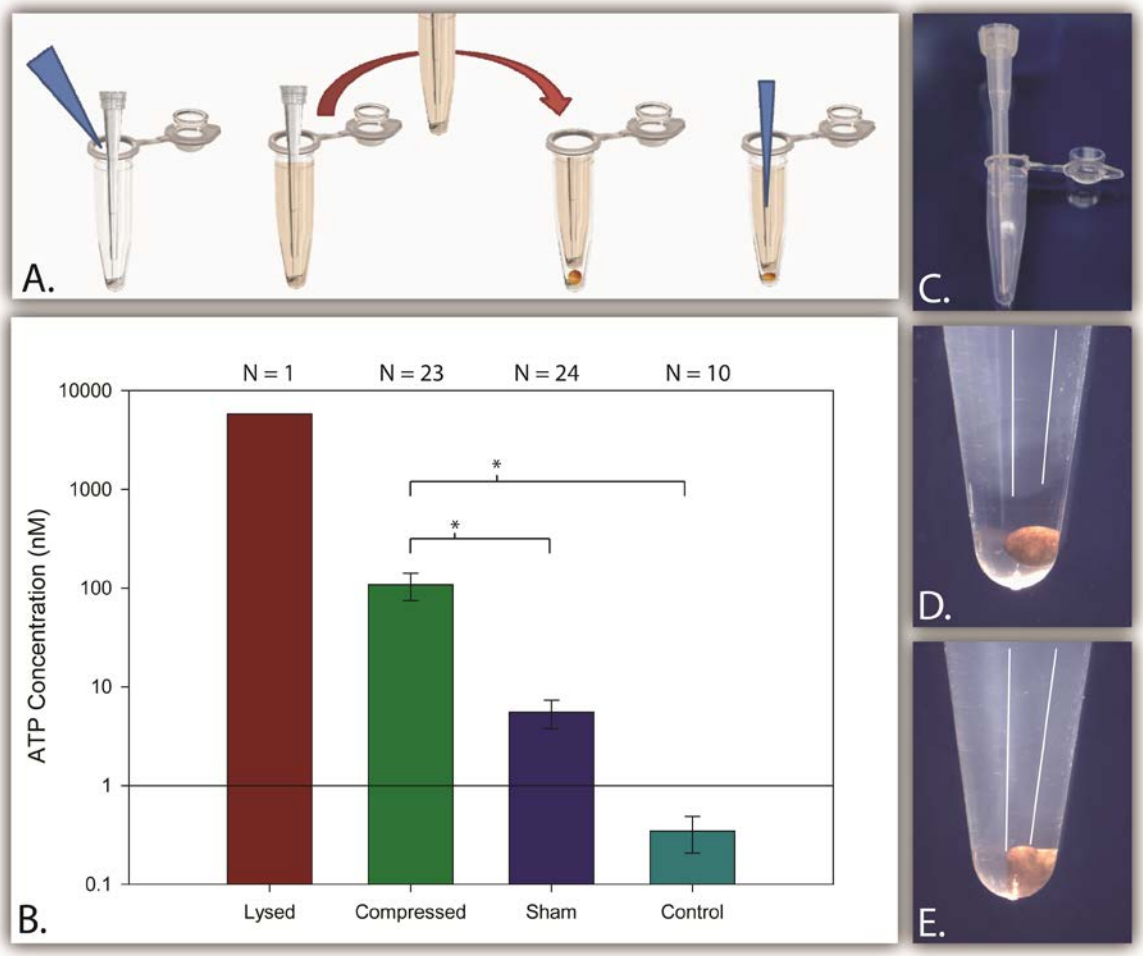

Figure 41. Embryos under compression release endogenous extracellular ATP.

The compression assay was conducted following the schematic laid out in (A). First, an agarose mold was created by placing a small micropipette tip in a PCR tube and then adding 280 uL of 1.5% agarose (C). Next, the agarose was cooled and allowed to harden, and then the tip was removed leaving a channel behind. The mold was then placed in a second PCR tube containing 70 uL of media with an embryo (D) and then used to gently compress the embryo for 1 minute (E). Media was collected through the channel and a high-sensitivity luciferase kit was used to estimate concentrations of extracellular ATP (B). Stars indicate a statistically significant difference, as calculated by an independent t-test, with $p < 0.05$.

A.1.3 Contractility induced by extracellular ATP is sufficient to increase tissue stiffness

Because ATP induces a contractile response in epithelia, and we have previously observed that other modulators of actomyosin contractility change bulk tissue mechanical properties (Zhou, Kim et al. 2010, Kim, Jackson et al. submitted), we speculated that ATP may increase tissue stiffness. Using a custom-built microaspiration rig, Stage 10 embryos were exposed to a light suction pressure while tissue displacement into the microchannel was measured (Fig. 42A, C). Using a power law creep compliance model for the tissue response, the fast response (t = 30s) and steady-state response (t = 300s) was calculated. We then inverted compliance values to obtain stiffness measurements (Fig. 42B). Using this method, we found a dose-dependent increase in stiffness in response to ATP. Control embryos had a stiffness measurement of approximately 10 Pa, while embryos incubated in 4uM ATP had a nearly 2-fold increase in stiffness compared to the control. Furthermore, 40uM ATP treated embryos were approximately 7-fold stiffer than controls. This revealed that inducing contractility with ATP is sufficient to increase bulk tissue stiffness, and thus may provide insight for how embryos endogenously modulate their mechanical properties.

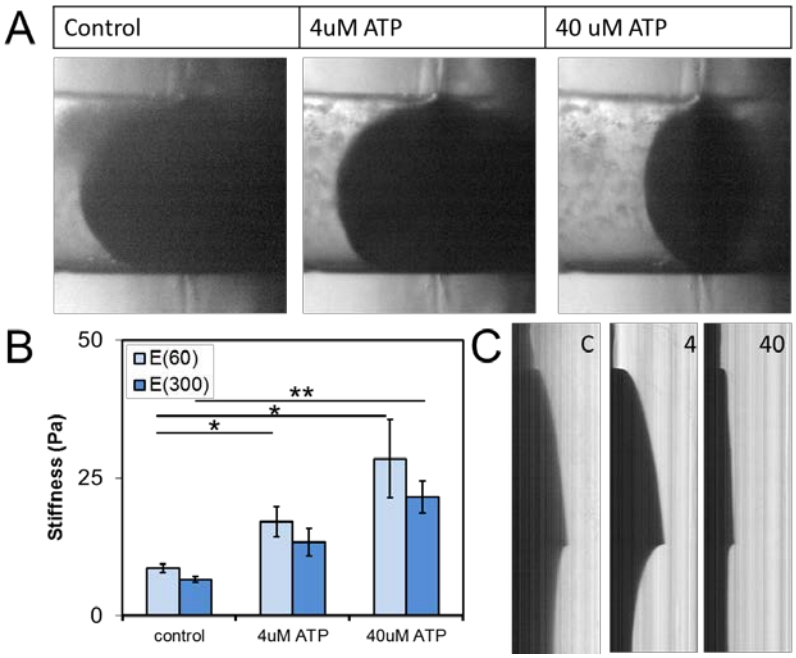

Figure 42. Embryos increase stiffness in response to extracellular ATP.

(A) Frame at 300s taken from microaspirator time lapse, showing time of max suction-induced tissue displacement. (B) Stiffness calculations for embryos in response to ATP. (C) Representative kymographs of microaspirator time lapses.

A.1.4 P2Y2R antisense knockdown abolishes contractile response to ATP

Having identified the receptor of ATP as P2Y2R, we wanted to first know expression patterns of P2Y2R in the *Xenopus* embryo. Using RNA *in situ* hybridization, we find that both alloalleles P2Y2A and P2Y2B are expressed in the Stage 10 and 11 animal cap (Fig. 43A) as well as throughout the ectoderm in the Stage 15 neurula (not shown). After confirming native expression of P2Y2R in the animal cap, 32 ng total of two morpholino oligonucleotides specific to alloalleles A and B (P2Y2R-MO) were injected into epithelial precursors at the 8-cell stage to knock down P2Y2R expression. After performing ATP nano-perfusion assays, we generated

kymographs to visualize the contraction (Fig. 43B) on linear region across the center of the embryo (Fig. XB; yellow line). Using the kymographs, embryos were categorized based on their response as strong (++), positive (+) or not contractile (-). We determined that only 30% of P2Y2R-MO injected embryos displayed a contractile response compared to almost 95% of the control morpholino (COMO) injected embryos (Fig. 43C). By subtracting and t-projecting consecutive images, we were able to visualize contractions across the epithelial surface (Fig. 43D). By selecting an intensity threshold deemed contractile, we were able to quantify the area of contractile response (Fig. 43E) and found almost 20% more contractile area in COMO embryos vs. P2Y2-MO embryos. This strongly suggests that ATP induces epithelial contractility through the P2Y2R GPCR.

We were also curious if actomyosin structure was altered due to P2Y2R knockdown. We expressed COMO or P2Y2-MO in animal caps and fixed and stained them for actin, allowing us to quantify actomyosin based on intensity (Fig. 44). We find that in the animal cap, superficial ectoderm cells expressing P2Y2R-MO had a 10% reduction in actin compared to uninjected cells. This disparity only became more pronounced once P2Y2R-MO expressing cells were exposed to ATP, which caused an almost 40% difference in actin intensity between P2Y2-MO expressing and uninjected cells. Interestingly, the COMO injected cells also saw a reduction in actin when exposed to ATP, at about 10% compared to uninjected cells. Together, this evidence points to a role for ATP and P2Y2R in regulating epithelial contractility through direct effects on actin.

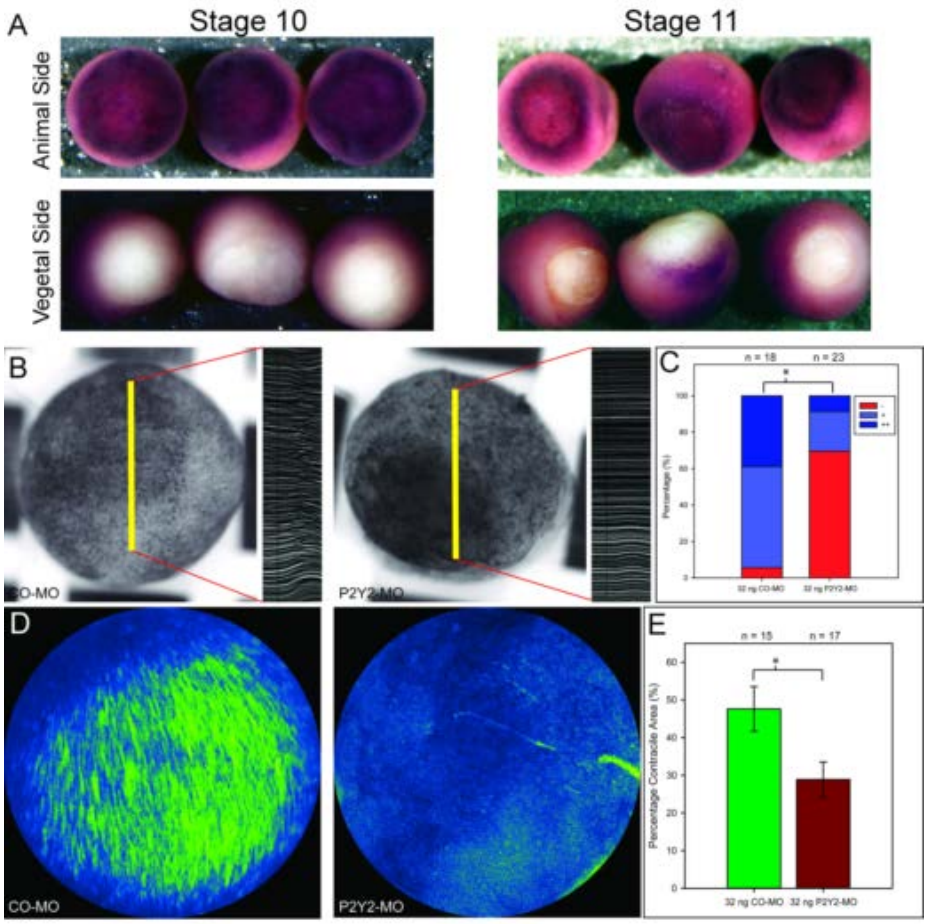

Figure 43. P2Y2R antisense knockdown reduces epithelial contractility.

(A) Embryos injected with either a 32 ng of a control morpholino oligomer (CO-MO) or 16 ng each of two morpholino oligomers corresponding to the two alloalleles of P2Y2R (P2Y2-MO) were subjected to a nanoperfusion assay with 4nL of 400 uM ATP. Both kymograph analysis and a contraction heat map analysis indicate a significant decrease in epithelial contractility due to P2Y2R antisense knockdown. (A) Representative kymograph of an embryo injected with CO-MO and (B) P2Y2-MO. (C) Responses to nanoperfusion were categorically defined as either a strong contraction (++), some contraction (+), or little to no contraction (-) based on kymographs. Significance determined by ANOVA, $p < 0.001$. (D) Representative contraction heat map of an embryo injected with CO-MO and (E) P2Y2-MO. (F) Quantification of the percentage of contractile area on the embryonic epithelium showed a significant difference as determined by an independent t-test, $p < 0.05$.

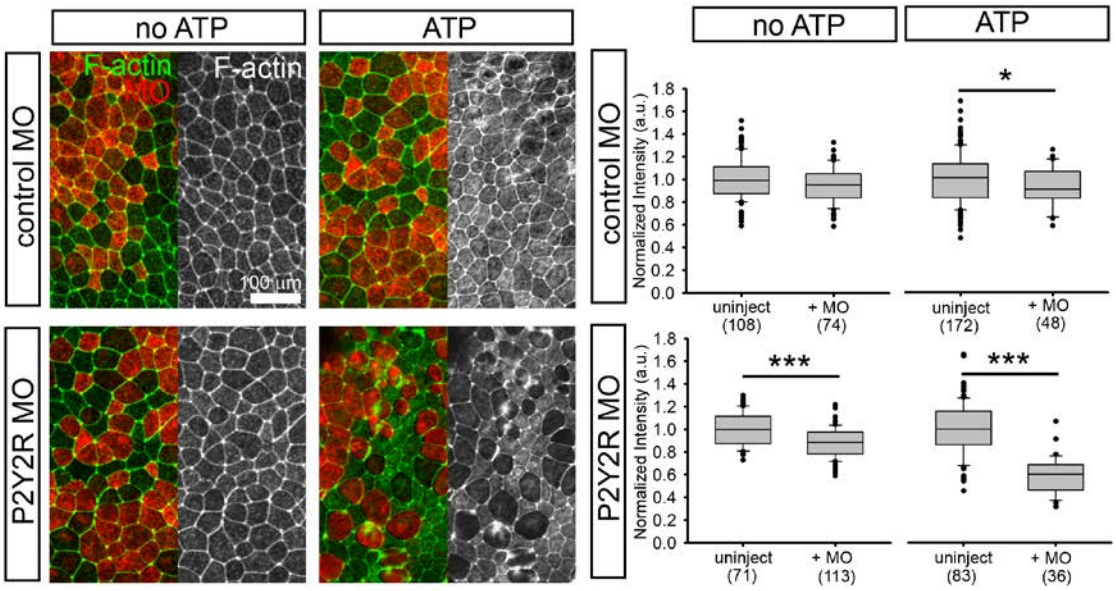

Figure 44. P2Y2R antisense knockdown reduces actomyosin in cells.

Left panels show F-actin stained (green) and co-injected rhodamine dextran identifying morpholino expressing cells (red) alongside grayscale image of F-actin channel. Right panels show normalized actin intensity differences between uninjected cells and morpholino expressing cells. Image analysis performed by Lance Davidson.

A.1.5 Antisense knockdown can be rescued by overexpression of P2Y2R

To confirm the specificity of our antisense knockdown for P2Y2R, we synthesized mRNA for the alloallele P2Y2A containing a silent mutation that would prohibit our morpholino from binding to it. Then we injected embryos with both the P2Y2-MO, co-injected with rhodamine dextran (RDA), and the P2Y2A mRNA, co-injected with fluorescein dextran (FDA; Fig. 45A). We note that most embryos develop normally, with some cases of exogastrulation developing in P2Y2A overexpression embryos, similar to embryos overstimulated with ATP. By performing nano-perfusions, we were able to establish that expressing P2Y2A mRNA in P2Y2-

MO injected embryos successfully rescued the ATP-induced contractility (Fig. 45B-C). Approximately 90% of embryo injected with P2Y2A mRNA and P2Y2-MO exhibited epithelial contractility in response to ATP, compared to 95% in COMO injected and 70% in P2Y2A mRNA and COMO injected (Fig. 45B). By contrast, only about 25% of P2Y2-MO injected embryos displayed a contractile response. Similar results were seen when calculating the percent contractile area (Fig. 45C): while P2Y2-MO significantly reduced the contractile area compared to the control, P2Y2A mRNA was able to rescue the contractile response (Fig. 45C). Thus, we verified that the P2Y2-MO was in fact knocking down P2Y2R to disrupt actin signaling.

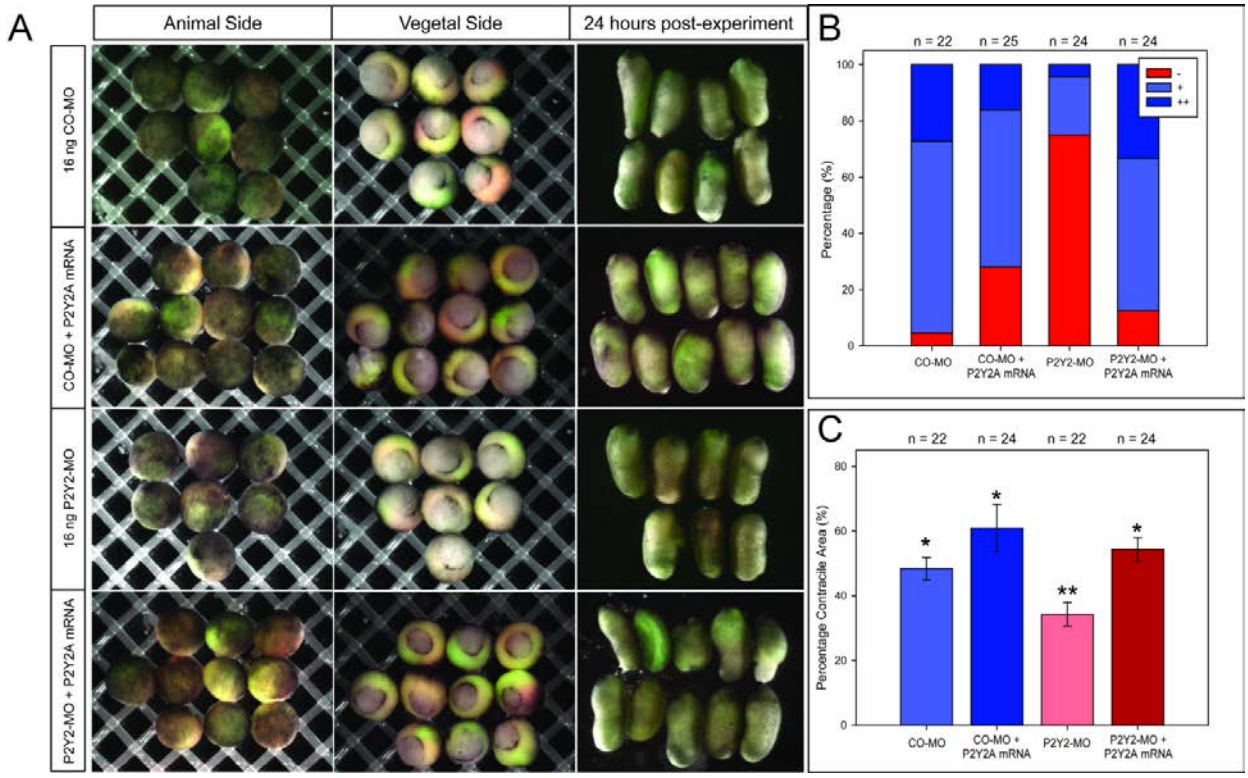

Figure 45. P2Y2A mRNA can "rescue" epithelial contractility decreases due to P2Y2R antisense knockdown.

(A) Embryos were first injected with either a 16 ng of a control morpholino oligomer (CO-MO) or 8 ng each of two morpholino oligomers corresponding to the two alloalleles of P2Y2R (P2Y2-MO). Then they were injected with either 8 nL of P2Y2A mRNA or a control solution. Morpholino oligomers were co-injected with rhodamine dextran and the mRNA was co-injected with fluorescein dextran, to identify areas of expression. Embryos rarely showed developmental defect. (B) Contractility analysis by kymograph or (C) by percentage contractile area demonstrated that while P2Y2R antisense knockdown significantly decreased contractility, P2Y2A mRNA would could "rescue" that effect to produce embryos not significantly different than the control.

Next we wondered if the bulk tissue stiffness was affected by antisense knockdown of P2Y2R. We used microaspiration to measure the mechanical properties of COMO-, P2Y2-MO- and P2Y2A mRNA-injected embryos (Fig. 46). P2Y2-MO embryos were significantly less stiff than control embryos, by a factor of almost 30%. Moreover, by injecting the P2Y2A mRNA with P2Y2-MO, we were able to rescue the effects of the morpholino, instead increasing stiffness by approximately two-fold compared to the control. We also note a slight, although not statistically significant, increase in stiffness due to expression of P2Y2A mRNA alone, perhaps suggesting that overexpression of P2Y2R leads to a "hyper-contractile" phenotype to increase mechanical properties. We therefore conclude that ATP-P2Y2R signaling is sufficient to regulate the stiffness of the gastrulating embryo, implying a role of ATP induced actomyosin contractility for maintaining the robustness of mechanical properties to ensure proper gastrulation.

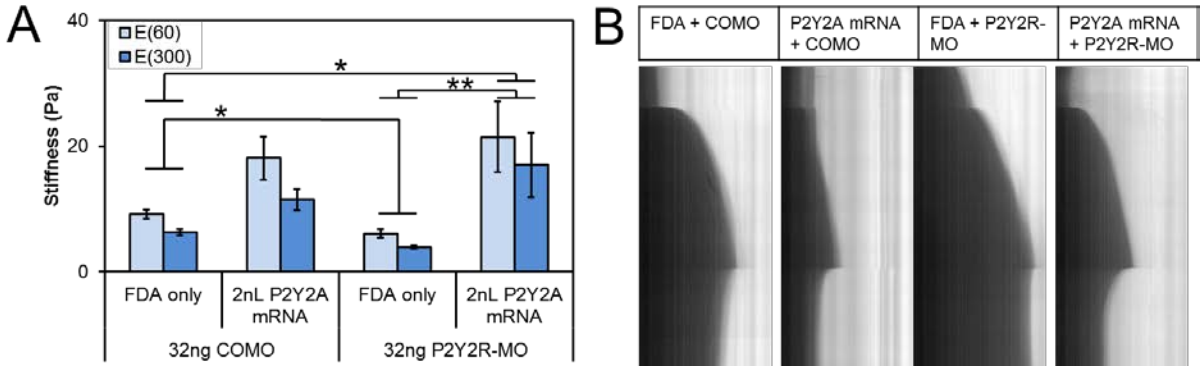

Figure 46. Antisense knockdown of P2Y2R reduces stiffness and can be rescued by over expression of P2Y2A mRNA.

(A) Bulk tissue stiffness calculations for Stage 10 embryos shows reduced compliance due to P2Y2R-MO, which is rescued by overexpression of P2Y2A mRNA. (B) Representative kymographs of microaspiration experiments.

A.2 DISCUSSION

Actomyosin contractility has a fundamental role in morphogenesis, including dorsal closure in Drosophila (Mammoto and Ingber 2010), convergent extension and neural tube folding (Moore, Keller et al. 1995, Wallingford and Harland 2001, Davidson, Marsden et al. 2006, Zhou, Kim et al. 2009) and heart tube formation and looping (Shi, Varner et al. 2015). The process of development requires cell to dynamically remodel their cytoskeleton in order to change shape and phenotype (Kim, Jackson et al. 2016), create and maintain junctions and form complex tissue-scale shapes and structures. As we previously reported the capability of cell lysate to induce robust contractions when perfused over embryonic epithelial (Joshi, von Dassow et al. 2010), we now identify the ATP-P2Y2R signaling pathway in operating those contractile responses.

We propose ATP-P2Y2 induced contractility to be a unique case of a mechanical feedback loop required for regulating gastrulation. Endogenous extracellular ATP likely activates P2Y2R at a basal level to control the speed of epiboly, to ensure robust development. If the embryo encounters a mechanical disturbance, that in turn induces higher release of extracellular ATP to stimulate P2Y2R-controlled actomyosin contractility, to perhaps increase the stiffness in the embryo and in turn minimize the negative consequences of the disturbance. Once the embryo begins to return to homeostasis, it likely releases ecto-ATPases to return to basal activity. Further studies will need to investigate the downstream effectors of P2Y2, to establish how the GPCR signaling can stimulate contractility.

A.3 EXPERIMENTAL PROCEDURES

A.3.1 Compression assays

Embryos were placed into PCR tubes containing 1/3X MBS. Molds composed of 1.5% agarose in a separate PCR tube and a micropipette tip was placed in the center in order to facilitate media extraction. The agarose mold was then moved into the PCR tube containing the embryo and the lid was closed in order to apply a light compressive force on the embryo for approximately 5 minutes. Then using a micropipette, approximately 20µL of solution was extracted from channel in the center of the mold. That media was then analyzed with the ATP Bioluminescence Assay Kit HS II (Roche) using manufacturer's instructions.

A.3.2 Nano-perfusion assays and image analysis

Embryos were cultured until Stage 10 in 1/3X MBS and placed on a nylon mesh to stabilize them for filming. A microinjection rig (Harvard Apparatus PLI-90, Holliston MA) was utilized utilized to deliver 4 nl of 40uM ATP mixed with trace amounts of black non-waterproof ink (Higgins Fountain Pen India Ink; Utrecht Art, Cranbury NJ) for visualization purposes. Time-lapse sequences of perfusion experiments were recorded by first obtaining a baseline for the embryo, performing the perfusion and then image at one frame per second for 2 minutes using a CCD camera (Scion Co, Frederick,MD) mounted on a dissecting stereomicroscope.

To generate kymographs, a linear region spanning the center of the embryo was selected and mapped over the timelapse using ImageJ (Schneider, Rasband et al. 2012). Contrast was enhanced and each kymograph was manually categorized into strong contraction, contraction or no contraction. To quantify contractile area, an ellipse was fit to the shape of the embryo and outside pixels were set to zero. Then, consecutive images were subtracted and t-projected to obtain the contractile heat map. An arbitrary threshold value was first manually selected to highlight pixels that appeared contractile and then kept constant throughout analysis of the entire experimental set of images.

A.3.3 RNA *in situ* hybridization

Embryos were devitellinized and fixed at Stage 10 or 11 with MEMFA (3.6% PFA supplemented with 1x MEM salt solution). Embryos were rehydrated and permeabilized before fixed RNAs were hybridized with a digoxigenin-labeled probe for P2Y2A or P2Y2B. Hybridization was carried out overnight in a 60C shaking water bath. Following washing and a incubation with 2% blocking solution, embryos were then incubatd with an anti-digoxigenin antibody overnight at 4C. Finally, after washing, embryos were subjected to a chromogenic reaction until color began to show, approximately 1 hour.

A.3.4 Stiffness measurements using microaspriation

Microaspiration was performed using a custom dual-reservoir microaspirator apparatus previous described (Sections 3.4.4, 4.4.4, 5.4.4) and illustrated in Fig. 20. In brief, a suction pressure was applied to the dorsal marginal zone of the embryo for 300 seconds as displacement

into a microchannel was tracked. From the displacement data, a power law model for creep compliance implemented in MATLAB (Mathworks) was used to obtain time-dependent compliance measurements for each embryo. Compliance values were inverted to obtain stiffness values depicted in graphs.

A.3.5 Extracellular ATP measurements

A mold composed of approximately 280uL of 1.5% agarose was cast in a PCR tube containing a micropipette tip. Once the agarose set, tip was removed and the PCR tube was cut off. The agarose mold was then placed into a second PCR tube containing an embryo and the lid of the PCR tube was closed to exert slight compression onto the embryo. If the embryo was wounded in any way, that sample was not used for calculations.

After applying the compression for approximately 5 minutes, media was extracted through the channel in the center of the mold and the ATP concentrations in the media were determined with the ATP Bioluminescence Assay Kit HS II (Roche) according to manufacturer's instructions. A serial dilution of known ATP concentration solutions was used to calibrate luciferase luminescence measurements, and then measurements of extracted data was fit to the log-log curve.

APPENDIX B

PROTOCOLS AND IMAGE ANALYSIS

This appendix contains a number of protocols, including the intravital nuclei tracking (Section B.1), low-resolution traction force microscopy (Section B.2), tissue deformation tracking of the ventral heart forming region (Section B.3) and finally, YAP colocalization in mesenchymal cell aggregates (B.4). In addition to the protocols, the code I used for the image analysis is also included.

B.1 NUCLEI TRACKING

The following protocol outlines the microsurgical preparation, imaging and image analysis for the intravital nuclei tracking of HPCs.

B.1.1 Microsurgery and Imaging Protocol

1.) Inject embryos with an mRNA encoding a fluorescently tagged nuclear label, e.g. H2B-mCherry. This was most frequently done at the 1-cell stage, targeting expressing to the mesoderm (vegetal injections).
    a. After much trial and error, I found that mCherry was preferable to RFP or GFP because of its photostability and because of the likelihood of the 488nm wavelength laser required for imaging GFP killing samples over long term imaging. Additionally, if using the nkx2.5-GFP line of transgenic embryos, mCherry has less spectral overlap than RFP.
2.) Culture embryos to Stage 12 or 13. The rest of the protocol takes some time before imaging will start, which I typically began around Stage 17.
3.) Prepare DFA supplemented with antibiotic and a plate of 1.5% agarose to perform microsurgery. Also prepare a fibronectin coated coverslip, by incubating coverslip with a 10μg/mL solution of fibronectin.
    a. Note this is half the concentration of what is typically used in *Xenopus* explant assays. Again, I found through trial and error that HPC movements are better facilitated on lesser concentrations.
4.) Use forceps to dig well in agarose plate where embryos can be stabilized. Once devitellinized, carefully remove the superficial and deep ectoderm layers from the ventral region. Microsurgery will proceed as outlined in the schematic in Fig. 47A.
    a. The window does not need to be large, as it will expand (Fig. 47B). However, I found it very important to actually REMOVE cells, and not just perform a cut and leave ectoderm "hanging," which almost always results in the window closing.

Sometimes, it was easier to remove ectoderm already "hanging" with forceps rather than hair tools.

5.) As soon as the window is cut, move the embryo onto the fibronectin coverslip, where the chamber should now contain DFA, and position the embryo so that the window is down on fibronectin. Gently compress the embryo under a coverslip.
    a. Gentle is the key word here! Compressing too hard will result in HPC movements not proceeding correctly (it's not clear if this is due to inhibited endodermal movements or accelerated MET or something else). However, compressing too lightly will result in the window not adhering and thus closing. Trial and error will likely be needed to find the "sweet spot."
6.) After 2 hours, remove the coverslip. Letting the coverslip remain will hinder HPC movements. Removing the coverslip sooner will not give the window enough time to adhere.
7.) Once finished and sealing the chamber, proceed to imaging.
    a. Imaging requires balancing the need of a frequent enough frame rate, obtaining adequate image contrast and minimizing phototoxicity to the embryo. Turn down laser intensity as much as possible. Doing at least two line averages is necessary to minimize noise for segmentation purposes in the analysis. I found that imaging once every 15-20 minutes was sufficient for tracking.

B.1.2 Image analysis

1.) For image analysis, I wrote an image analysis algorithm (Section B.1.3) utilizing a finite-element (FEM) based registration filter, the logic of which is explained in Fig. 47C.

a. In brief, nuclei are segmented using a watershed method where the user selects the number of slices to analyze and the level used by the registration filter (typical level value choices range from 0.1 to 0.5). After entering a choice for the level, the user can check the first binary image to see if they want to change the level amount or proceed.
b. Once the nuclei are segmented, a centroid is calculated for each nucleus $C = (Cx_f, Cy_f)$.
c. Then, the FEM-based registration filter will calculate a transformation between the first image (fixed image) and the second image (moving image) to get a displacement function ($d(x,y)$). The displacement calculated for the centroid in the fixed image will be applied then be applied to serve as a search point for the corresponding centroid in the moving image $C' = (C + d(C))$.
d. The moving image is then searched with a given radius (r; set by variable minDistance and equal to 10 pixels in the provided code) to find a corresponding centroid that is the likely match.
    i. If more than one centroid falls in that neighborhood, the one with the minimum distance is chosen.
e. Once a match is determined for a moving image centroid, it is not allowed to be matched again.
f. If a match is not found for a centroid in the fixed image, then the guess point continues to accumulate displacements for up to 5 more frames, e.g. $C'' = (C+d(C) + d'(C+d(C))$ and $C''' = C+d(C) + d'(C+d(C)+d''[C+d(C) + d'(C+d(C)]$ and so on, and matches continue to be attempted. After 6 consecutive match

failures, that centroid is removed from the list of potential matches and not attempted again.

g. If using a mask, the algorithm will search the first frame for which nuclei fall into that mask.

2.) The final output of the nuclei tracking is .txt file that was designed to be opened in Excel. Once open in Excel, it is recommended that the data be sorted by GlobalLabel and then by Frame. If using a mask, any nuclei not found in frame 0 should be deleted, as they were not found in the first masked frame.

3.) To obtain parameters to assess cell migration, a separate Mathematica code was used to analyze the .txt file (Section B.1.4). Parameters include:

a. Collectiveness: for a given nucleus, cells within a defined distance, i.e. neighborhood, were included in an analysis to determine the average cosine similarity score of cell movement trajectories.

b. Persistence: for a given nucleus, the effective distance moved in five frames over the total distance moved.

c. Speed: distance per frame

d. Percent co-aligned: % of cells with a collectiveness score > 0.7.

The Mathematica program takes the input file (input.xlsx) and outputs correlation and persistence data into separate .xlsx files (outputCorrelation.xlsx and outputPersistence.xlsx) and prints a graph of total displacements (graph.tif). Microsoft Excel was ultimately used to graph the collectiveness and persistence data as Mathematica graph lack presentation quality.

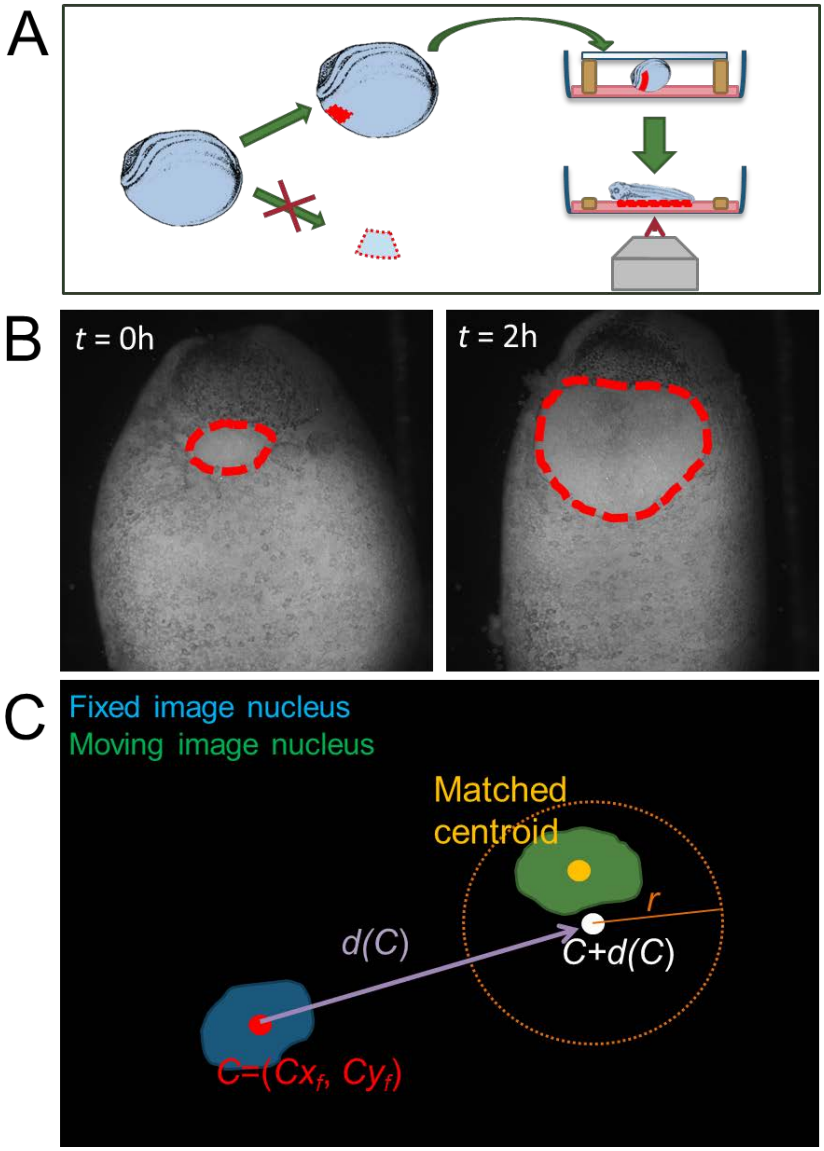

Figure 47. Microsurgical preparation and image analysis for intravital nuclei tracking

(A) Schematic for removing ventral ectoderm (blue) to enable viewing into the mesoderm (red), mounting embryo window onto fibronectin (pink) underneath a glass coverslip (white) mounted with silicone grease (brown). The coverslip is removed and imaging proceeds from below. (B) Example of how a small window (red outline) expands greatly over 2h. (C) Schematic for nuclei tracking algorithm logic. See text for details.

B.1.3 Nuclei tracking code

The following code was written in C++ using the open-source libraries of ITK.

```
//including all necessary headers:
#include "itkImage.h"
#include <iostream>
#include <stdio.h>
#include <string>
#include <math.h>
#include "itkImageRegionConstIterator.h"
#include "itkImageRegionIterator.h"
#include "itkImageRegionIteratorWithIndex.h"
#include "itkImageFileReader.h"
#include "itkImageFileWriter.h"
#include "itkTIFFImageIO.h"
#include "itkLineIterator.h"
#include "itkImageSeriesWriter.h"
#include "itkStatisticsImageFilter.h"

#include "itkRescaleIntensityImageFilter.h"
#include "itkHistogramMatchingImageFilter.h"
#include "itkFEMRegistrationFilter.h"
#include "itkExtractImageFilter.h"
#include "itkMaskImageFilter.h"
#include "itkConnectedThresholdImageFilter.h"
#include "itkCurvatureFlowImageFilter.h"
#include "itkCastImageFilter.h"
#include <itkRGBPixel.h>

//***for binary to label map
#include "itkBinaryImageToLabelMapFilter.h"
#include "itkLabelMapToLabelImageFilter.h"
#include "itkGrayscaleDilateImageFilter.h"
#include "itkLabelGeometryImageFilter.h"
#include "itkBinaryImageToShapeLabelMapFilter.h"
#include "itkBinaryFillholeImageFilter.h"
#include "itkShapeOpeningLabelMapFilter.h"
#include "itkLabelStatisticsImageFilter.h"

//**************************for segmentation
#include "itkShapeDetectionLevelSetImageFilter.h"
#include "itkCastImageFilter.h"
#include "itkGradientMagnitudeRecursiveGaussianImageFilter.h"
#include "itkSigmoidImageFilter.h"
```

```
#include "itkFastMarchingImageFilter.h"
#include "itkBinaryThresholdImageFilter.h"
#include "itkSimilarityIndexImageFilter.h"

#include "itkScalarToRGBPixelFunctor.h"
#include "itkUnaryFunctorImageFilter.h"
#include "itkVectorCastImageFilter.h"
#include "itkVectorGradientAnisotropicDiffusionImageFilter.h"
#include "itkWatershedImageFilter.h"
#include "itkRescaleIntensityImageFilter.h"
#include "itkScalarToRGBColormapImageFilter.h"

#include "itkLabelImageToLabelMapFilter.h"
#include "itkLabelImageToShapeLabelMapFilter.h"

//*********************for binary
#include "itkBinaryThresholdImageFilter.h"

//***for binary to image
#include "itkBinaryImageToLabelMapFilter.h"
#include "itkLabelObject.h"

//*************************for pointset
#include "itkPointSet.h"

//*************************for drawing lines
#include "itkLineIterator.h"



//Namespace std will make it easier to use cout and cin functions
using namespace std;

//Defining types globally for easy access and troubleshooting
//~~~
```

```
typedef itk::Image< float, 3 >
       FloatImageType3D;
typedef itk::ImageFileReader<FloatImageType3D>                FloatReaderType3D;

//~~~
```

```
typedef itk::BinaryThresholdImageFilter<FloatImageType, ImageType2D >
                    BinaryType;
typedef itk::BinaryImageToLabelMapFilter<ImageType2D>
                           BinaryImageToLabelMapFilterType;

typedef itk::Image<unsigned long, 2>   LabeledImageType;
typedef itk::Image<unsigned short,2>   ShortImageType;
typedef itk::LabelImageToShapeLabelMapFilter <ShortImageType>
LabelImageToLabelMapFilterType;


//function
prototypes~~~
```

```
                                                        int maxNucSize, int minNucSize,
float minRoundness);

int main()
{
       double pi = 3.14159265359;
       //First I will make a variable for the input filename, which will be put in at
command line using the cin function
       char* inputFilename = new char[50];

       cout << "Please enter the stack's file name (must be a TIFF file)." << endl;
       cout << "WARNING:: Entering more than 50 characters will cause an error." << endl;
       cout << "Input stack file name: ";
       cin >> inputFilename;

       stringstream inputNames;
       inputNames << inputFilename << ".tif" ;

       stringstream maskNames;
       maskNames << inputFilename << "_Mask.tif" ;

       ReaderType3D::Pointer reader = ReaderType3D::New();
                              //creating the reader
    reader->SetFileName( inputNames.str().c_str() );
                     //the reader will now look for the file name input by the user
       ReaderType2D::Pointer maskReader = ReaderType2D::New();

    maskReader->SetFileName( maskNames.str().c_str() );

       try
    {
                     // reader will attempt to update with inputted filename
              maskReader->Update();
    }
       catch (itk::ExceptionObject &e)
    {
              cerr << "Mask image not found." << endl << e << endl;
                                    // if file cannot be found, program will exit
              return EXIT_FAILURE;
    }

       itk::ImageIOBase::Pointer imageIO =
itk::ImageIOFactory::CreateImageIO(inputNames.str().c_str(),
itk::ImageIOFactory::ReadMode);
       if( !imageIO )
       {
              std::cerr << "Could not CreateImageIO for: " << inputFilename << std::endl;
              return EXIT_FAILURE;
       }
       imageIO->SetFileName( inputNames.str().c_str() );
       imageIO->ReadImageInformation();

       typedef itk::ImageIOBase::IOComponentType ScalarPixelType;
       std::cout << "image info: " << imageIO << std::endl; // '8'

       //typedef itk::VectorImage<unsigned short, 3>          VectorImageType;
       //typedef itk::Image<unsigned short, 3>                ScalarImageType;
       ImageType3D::SizeType inputSize;
```

```
        inputSize[0]=imageIO->GetDimensions(0);
        inputSize[1]=imageIO->GetDimensions(1);
        inputSize[2]=imageIO->GetDimensions(2);

        ImageType3D::IndexType start;
        start.Fill(0);
        FloatImageType3D::RegionType inputRegion(start,inputSize);

        cout << "Files found." << endl;

        BinaryImageToLabelMapFilterType::Pointer binaryImageToShapeLabelMapFilter =
BinaryImageToLabelMapFilterType::New();
        binaryImageToShapeLabelMapFilter->SetInput(maskReader->GetOutput());
        binaryImageToShapeLabelMapFilter->Update();

        BinaryImageToLabelMapFilterType::OutputImageType::LabelObjectType*  firstFrameMask
= binaryImageToShapeLabelMapFilter->GetOutput()->GetNthLabelObject(0);

        // Setting up the registration filter
        RegistrationType::Pointer registrationFilter = RegistrationType::New();
    registrationFilter->SetMaxLevel(1);
    registrationFilter->SetUseNormalizedGradient( true );
    registrationFilter->ChooseMetric( 0 );

        // Many of the following parameters were not changed from the example
registration.
        // I did change meshpixels around, but decided that 4 was the perfect balance of
speed vs. resolution
        // Number of iterations is 15 -- again to balance between speed and accuracy.
        // Elasticity and rho, while ITK recommended E/p to be roughly equal to 1, I found
that having a high elasticity and a low rho allowed for more meaningful strain maps.
        int meshPixels = 8;
        unsigned int maxiters = 20;
        float        E = 100;
        float        p = 1;
        registrationFilter->SetElasticity(E, 0);
        registrationFilter->SetRho(p, 0);
        registrationFilter->SetGamma(1., 0);
        registrationFilter->SetAlpha(1.);
        registrationFilter->SetMaximumIterations( maxiters, 0 );
        registrationFilter->SetMeshPixelsPerElementAtEachResolution(meshPixels, 0);
        // affects pixel resolution of strain maps.  low values EXTREMELY slow
        registrationFilter->SetWidthOfMetricRegion(1, 0);
        registrationFilter->SetNumberOfIntegrationPoints(2, 0);
        registrationFilter->SetDoLineSearchOnImageEnergy( 0 );
        registrationFilter->SetTimeStep(1.);
        registrationFilter->SetEmployRegridding(false);
        registrationFilter->SetUseLandmarks(false);

        // Create the material properties
        itk::fem::MaterialLinearElasticity::Pointer m;
        m = itk::fem::MaterialLinearElasticity::New();
        m->SetGlobalNumber(0);
        m->SetYoungsModulus(registrationFilter->GetElasticity()); // Young's modulus of
the membrane
        m->SetCrossSectionalArea(1.0);                            // Cross-sectional area
        m->SetThickness(1.0);                                     // Thickness
        m->SetMomentOfInertia(1.0);                               // Moment of inertia
```

```
	m->SetPoissonsRatio(0);
	// Poisson's ratio -- tissues are mostly composed of water, which is
incompressible.  So 0.5 makes sense here.
	m->SetDensityHeatProduct(1.0);                          // Density-Heat capacity
product

	// Create the element type
	ElementType::Pointer e1=ElementType::New();
	e1->SetMaterial(m.GetPointer());
	registrationFilter->SetElement(e1.GetPointer());
	registrationFilter->SetMaterial(m);


	//establishing variables for the input image's largest possible region and size

	//the size of the slices will be identical to the input images, except with a size
of 0 in the z-direction
	ImageType3D::SizeType desiredSize = inputSize;
	desiredSize[2] = 0;

	//this allows the user to choose how many slices to the program on, so you need
not run the entire registration (which takes a long time) if you
	//are debugging or adjusting parameters.  There is also a checkpoint to assure the
range is within the actual size of the image.
	int range;
	cout << endl << "Please enter how many slices you would like to test" << endl;
	cout << "Number of slices: ";
	cin >> range;
	if(range>inputSize[2])
	{
		cerr << "Range invalid.";
		return EXIT_FAILURE;
	}
	else
	{
		cout << "Range accepted" << endl;
	}

	//making a variable for the output data file's name, which can be input by the
user.  The file name is then put into a FILE* pointer to create, open and write the data.
	//The first line is written at this point, which contains all the headers of the
data.  The format may look strange (\t indicating tabs), but when opened in Microsoft
	//Excel, it is very well organized.

	stringstream dataNames;
	dataNames << inputFilename << "_outputData.txt";

	FILE * pFile2 = fopen (  dataNames.str().c_str(), "w" );
	fprintf(pFile2,
"Frame\tLabelInFixed\tLabelInMoving\tGlobalLabel\tX_Fixed\tY_Fixed\tX_Moving\tY_Moving\tD
istance\tTheta_Deg\tTheta_Rad\n");

	cout<<"Extracting..."<<endl;
	ExtractFilterType::Pointer sampleExtractor = ExtractFilterType::New();
	sampleExtractor->InPlaceOn();
	sampleExtractor->SetDirectionCollapseToSubmatrix();
	ImageType3D::IndexType extractorStart = inputRegion.GetIndex();
	extractorStart[2]=0;
```

```
	ImageType3D::RegionType desiredRegion;
	desiredRegion.SetSize( desiredSize );
	desiredRegion.SetIndex( extractorStart );
	sampleExtractor->SetExtractionRegion( desiredRegion );
	sampleExtractor->SetInput( reader->GetOutput() );

	cout<<"Smoothing..."<<endl;
	CurvatureFlowImageFilterType::Pointer smoothing =
CurvatureFlowImageFilterType::New();
	int iter = 5;
	float timeStep = 0.125;
	smoothing->SetNumberOfIterations( iter );
	smoothing->SetTimeStep( timeStep );
	smoothing->SetInput( sampleExtractor->GetOutput() );

	cout<<"GradMag..."<<endl;
	float gradMagKernel = 2.0;
	typedef itk::GradientMagnitudeRecursiveGaussianImageFilter< FloatImageType,
FloatImageType > GradientImageType;
    GradientImageType::Pointer gradMagnitude = GradientImageType::New();
    gradMagnitude->SetInput( smoothing->GetOutput() );
    gradMagnitude->SetSigma( gradMagKernel );
	gradMagnitude->Update();

	cout<<"Stats..."<<endl;
	typedef itk::StatisticsImageFilter<FloatImageType> StatisticsImageFilterType;
	StatisticsImageFilterType::Pointer statisticsImageFilter =
StatisticsImageFilterType::New ();
	statisticsImageFilter->SetInput(smoothing->GetOutput());
	statisticsImageFilter->Update();
	float min = statisticsImageFilter->GetMinimum();
	float max = statisticsImageFilter->GetMaximum();
	float mean = statisticsImageFilter->GetMean();
	float sigma = statisticsImageFilter->GetSigma();
	float range2 = max-min;
	float coeff;

	int maxNucleusSize = 310;
	float minRoundness = 0.8;
	int minNucSize = 20;
	float threshold = 0.005;
	float intensityThreshold = mean;
	float level;

	int inside = 255;
	int outside = 0;

	cout<<"Segmenting..."<<endl;
	ShortImageType::Pointer sampleSegmentation = ShortImageType::New();
	sampleSegmentation->SetRegions( gradMagnitude->GetOutput()-
>GetLargestPossibleRegion() );
	sampleSegmentation->Allocate();

	for(unsigned int j = 0; j < 1; j++)
	{

		cout << endl << "Please enter the desired level:" << endl;
		cin >> level;
```

```
		//establishes the range of thresholding it must pass
		sampleSegmentation = PerformSegmentation(smoothing-
>GetOutput(),gradMagnitude-
>GetOutput(),threshold,level,intensityThreshold,maxNucleusSize,minNucSize,minRoundness);

		stringstream binNames;
		binNames << inputFilename << "_FirstBinary" << ".tif";
		WriteShortImage(sampleSegmentation, binNames.str().c_str());

		cout << endl << "Please check that you are satisfied with the binary image"
<< endl;
		int oneForRepeat;
		cout << endl << "If you would like to change the threshold values, please
enter 1. To continue, enter 2" << endl;
		cin >> oneForRepeat;
		if(oneForRepeat == 1)
			{j=-1;}
	}

	RGBImageType::Pointer allDispImage = RGBImageType::New();
	allDispImage->SetRegions(sampleExtractor->GetOutput()-
>GetLargestPossibleRegion());
	allDispImage->Allocate();
	RGBPixelType empty;
	empty[0] = 0;
	empty[1] = 0;
	empty[2] = 0;
	allDispImage->FillBuffer(empty);

	//min nuclei size
	int nucleiSize = 6;
	float maxDistance = 10;
	int unmatchedTimeLimit  = 6;

	int xWindowMin=1;
	int yWindowMin=91;
	int xWindowMax=501;
	int yWindowMax=447;


	int vectorSize = 50000;
	vector< vector<int> > global(vectorSize);

	//nuclei that are not matched will be stored in unmatchedDisps with their global
label in pos 0, their most recently tracked x-coord in pos 1, and their most recently
tracked y-coord in pos 2.
	//if they continue to not be matched, displacements will be added on and placed
into pos 3 (x-coord) and pos 4 (y-coord).
	//pos 5 will contain a count of how many times it has not been matched
	vector< vector<double> > unmatchedDisps(vectorSize);

	for(int i=0; i < vectorSize ; i++)
	{
		global[i].push_back(i+1000);
		unmatchedDisps[i].push_back(i+1000);
		unmatchedDisps[i].push_back(-1);
		unmatchedDisps[i].push_back(-1);
```

```
                unmatchedDisps[i].push_back(-1);
                unmatchedDisps[i].push_back(-1);
                unmatchedDisps[i].push_back(0);

                for(int t=0; t < range; t++)
                {
                        global[i].push_back(-1);
                }
        }

        vector <int> listOfNucleiToMatch;

        // Finally the for loop, which will for iterative registrations, is setup.  The
range input by the user is used as the stopping point here.
        for(unsigned int j = 0; j < (range-1); j++)
        {

                //Cout number j so the user will know how far along their program is
                cout << endl << "Starting Registration #" << j << endl;

                // ExtractFilterType will allow for the extraction of 2D slices from the 3D
stack read in by the reader.  Two extractors are necessary
                // because the registration filter requires two images for a registration.
                // Initializing them now makes sense because these will be constant
throughout the registration loop.
                ExtractFilterType::Pointer extractor1 = ExtractFilterType::New();
                extractor1->InPlaceOn();
                extractor1->SetDirectionCollapseToSubmatrix();
                ExtractFilterType::Pointer extractor2 = ExtractFilterType::New();
                extractor2->InPlaceOn();
                extractor2->SetDirectionCollapseToSubmatrix();
                RGBExtractFilterType::Pointer RGBextractor1 = RGBExtractFilterType::New();
                RGBextractor1->InPlaceOn();
                RGBextractor1->SetDirectionCollapseToSubmatrix();
                RGBExtractFilterType::Pointer RGBextractor2 = RGBExtractFilterType::New();
                RGBextractor2->InPlaceOn();
                RGBextractor2->SetDirectionCollapseToSubmatrix();

                //Extractor1 will get slice j while extractor2 gets slice (j+1).  These can
then be plugged into the registration filter.
                ImageType3D::IndexType extractorStart1;
                extractorStart1.Fill(0);
                extractorStart1[2] = j;
                ImageType3D::RegionType desiredRegion1;
                desiredRegion1.SetSize( desiredSize );
                desiredRegion1.SetIndex( extractorStart1 );
                extractor1->SetExtractionRegion( desiredRegion1 );
                extractor1->SetInput( reader->GetOutput() );
                RGBextractor1->SetExtractionRegion( desiredRegion1 );
                RGBextractor1->SetInput( reader->GetOutput() );
                RGBextractor1->Update();


                ImageType3D::IndexType desiredStart2;
                desiredStart2.Fill(0);
                desiredStart2[2] = j + 1;
                ImageType3D::RegionType desiredRegion2;
                desiredRegion2.SetSize( desiredSize );
```

```
            desiredRegion2.SetIndex( desiredStart2 );
            extractor2->SetExtractionRegion( desiredRegion2 );
            extractor2->SetInput( reader->GetOutput() );
            RGBextractor2->SetExtractionRegion( desiredRegion2 );
            RGBextractor2->SetInput( reader->GetOutput() );
            RGBextractor2->Update();

            /*RescalerType::Pointer rescaler1 = RescalerType::New();
            RescalerType::Pointer rescaler2 = RescalerType::New();
            RescalerType::Pointer maskRescaler = RescalerType::New();
            rescaler1->SetInput(extractor1->GetOutput() );
            rescaler2->SetInput(extractor2->GetOutput() );
            const double desiredMinimum =  0.0;
            const double desiredMaximum =  255.0;
            rescaler1->SetOutputMinimum( desiredMinimum );
            rescaler1->SetOutputMaximum( desiredMaximum );
            rescaler1->UpdateLargestPossibleRegion();
            rescaler2->SetOutputMinimum( desiredMinimum );
            rescaler2->SetOutputMaximum( desiredMaximum );
            rescaler2->UpdateLargestPossibleRegion();*/

            CurvatureFlowImageFilterType::Pointer smoothing1 =
CurvatureFlowImageFilterType::New();
            CurvatureFlowImageFilterType::Pointer smoothing2 =
CurvatureFlowImageFilterType::New();

            smoothing1->SetNumberOfIterations( iter );
            smoothing1->SetTimeStep( timeStep );
            smoothing1->SetInput( extractor1->GetOutput() );
            smoothing2->SetNumberOfIterations( iter );
            smoothing2->SetTimeStep( timeStep );
            smoothing2->SetInput( extractor2->GetOutput() );
            GradientImageType::Pointer gradMagnitude1 = GradientImageType::New();
            gradMagnitude1->SetInput( smoothing1->GetOutput() );
            gradMagnitude1->SetSigma( gradMagKernel );
            GradientImageType::Pointer gradMagnitude2 = GradientImageType::New();
            gradMagnitude2->SetInput( smoothing2->GetOutput() );
            gradMagnitude2->SetSigma( gradMagKernel );

            ShortImageType::Pointer segmentation1 = ShortImageType::New();
            segmentation1->SetRegions( gradMagnitude1->GetOutput()-
>GetLargestPossibleRegion() );
            segmentation1->Allocate();
            ShortImageType::Pointer segmentation2 = ShortImageType::New();
            segmentation2->SetRegions( gradMagnitude2->GetOutput()-
>GetLargestPossibleRegion() );
            segmentation2->Allocate();

            StatisticsImageFilterType::Pointer minMax1 = StatisticsImageFilterType::New
();
            minMax1->SetInput(smoothing1->GetOutput());
            minMax1->Update();
            float min1 = minMax1->GetMinimum();
            float max1 = minMax1->GetMaximum();
            float mean1 = minMax1->GetMean();
            float sigma1 = minMax1->GetSigma();
```

```
		StatisticsImageFilterType::Pointer minMax2 = StatisticsImageFilterType::New
();
		minMax2->SetInput(smoothing2->GetOutput());
		minMax2->Update();
		float min2 = minMax2->GetMinimum();
		float max2 = minMax2->GetMaximum();
		float mean2 = minMax2->GetMean();
		float sigma2 = minMax2->GetSigma();

		segmentation1 = PerformSegmentation(smoothing1->GetOutput(),
gradMagnitude1->GetOutput(), threshold, level, mean1, maxNucleusSize, minNucSize,
minRoundness);
		segmentation2 = PerformSegmentation(smoothing2->GetOutput(),
gradMagnitude2->GetOutput(), threshold, level, mean2, maxNucleusSize, minNucSize,
minRoundness);

		LabelImageToLabelMapFilterType::Pointer binaryImageToLabelMapFilter1 =
LabelImageToLabelMapFilterType::New();
		LabelImageToLabelMapFilterType::Pointer binaryImageToLabelMapFilter2 =
LabelImageToLabelMapFilterType::New();
		binaryImageToLabelMapFilter1->SetInput(segmentation1);
		binaryImageToLabelMapFilter1->Update();
		binaryImageToLabelMapFilter2->SetInput(segmentation2);
		binaryImageToLabelMapFilter2->Update();

		int numNuclei1 = binaryImageToLabelMapFilter1->GetOutput()-
>GetNumberOfLabelObjects();
		int numNuclei2 = binaryImageToLabelMapFilter2->GetOutput()-
>GetNumberOfLabelObjects();

		vector<int> xCoord1;
		vector<int> yCoord1;
		vector<int> xCoord2;
		vector<int> yCoord2;
		vector<int> mapping1;
		vector<int> mapping2;
		vector<int> mappingGL;


		for(unsigned int i = 0; i < numNuclei1; i++)
		{
			FloatIndex center;
			center[0] = GetCenterOfMass(i,0,binaryImageToLabelMapFilter1);
			center[1] = GetCenterOfMass(i,1,binaryImageToLabelMapFilter1);
			int size = GetSize(i,1,binaryImageToLabelMapFilter1);

			int regionFlag;

			if(j==0)
			{
				cout<< "Checking nucleus #" << i << " of " << numNuclei1 <<
"..." << endl;
				regionFlag =
CheckIfIndexIsInsideMask(firstFrameMask,center[0],center[1]);
				if (regionFlag == 1)
				{
					cout << "Nucleus #" << i << " is inside the mask." <<
endl;
```

```
                        }
                }
                else
                {
                        regionFlag = 1;
                }
                //if(size > nucleiSize && center[0]>xWindowMin &&
center[0]<xWindowMax && center[1]>yWindowMin && center[1]<yWindowMax)
                if(size > nucleiSize && regionFlag == 1)
                {
                        xCoord1.push_back( center[0] );
                        yCoord1.push_back( center[1] );

                }
        }

        for(unsigned int i = 0; i < numNuclei2; i++)
        {
                FloatIndex center;
                center[0] = GetCenterOfMass(i,0,binaryImageToLabelMapFilter2);
                center[1] = GetCenterOfMass(i,1,binaryImageToLabelMapFilter2);
                int size = GetSize(i,1,binaryImageToLabelMapFilter2);

                //if(size > nucleiSize && center[0]>xWindowMin &&
center[0]<xWindowMax && center[1]>yWindowMin && center[1]<yWindowMax)
                if(size > nucleiSize)
                {
                        xCoord2.push_back( center[0] );
                        yCoord2.push_back( center[1] );
                }
        }

        // The histogram filter helps match the images -- it was taken from the
example registration provided in the ITK documentation
        // Parameters for the historgram filter have not been changed from the
example -- these seemed to work well.
        HEFilterType::Pointer IntensityEqualizeFilter = HEFilterType::New();
        IntensityEqualizeFilter->SetReferenceImage( extractor1->GetOutput() );
        IntensityEqualizeFilter->SetInput( extractor2->GetOutput() );
        IntensityEqualizeFilter->SetNumberOfHistogramLevels(100);
        IntensityEqualizeFilter->SetNumberOfMatchPoints(15);
        IntensityEqualizeFilter->ThresholdAtMeanIntensityOn();
        IntensityEqualizeFilter->Update();

        // Setting the images for registration filter and running the registration
        registrationFilter->SetFixedImage( extractor1->GetOutput() );
        registrationFilter->SetMovingImage(IntensityEqualizeFilter->GetOutput());
        registrationFilter->Modified();
        registrationFilter->RunRegistration();

        // The X- and Y- displacements from the registration need to be written and
stored.  deltaX and deltaY are initialized to do so.  the initial displacement image is
also initialized
        DispImage::Pointer displacement = DispImage::New();
        displacement = registrationFilter->GetDisplacementField();
        FloatImageType::Pointer deltaX = FloatImageType::New();
        FloatImageType::Pointer deltaY = FloatImageType::New();
        deltaX->SetRegions(extractor1->GetOutput()->GetLargestPossibleRegion());
```

```
            deltaX->Allocate();
            deltaY->SetRegions(extractor1->GetOutput()->GetLargestPossibleRegion());
            deltaY->Allocate();

            // the displacement is output as a vector image, so I will use iterators to
isolate the X- and Y-displacements.
            // as the iterator moves through the displacement images, it sets the value
of deltaX to the x component of the vector and deltaY to the y
            // component of the vector.
            FloatIteratorType itX(  deltaX, extractor1->GetOutput()-
>GetLargestPossibleRegion() );
            FloatIteratorType itY(  deltaY, extractor1->GetOutput()-
>GetLargestPossibleRegion() );
            DispIteratorType itD ( registrationFilter->GetDisplacementField(),
registrationFilter->GetDisplacementField()->GetLargestPossibleRegion());
            for(itX.GoToBegin(), itY.GoToBegin(), itD.GoToBegin(); !itD.IsAtEnd();
++itX, ++itY, ++itD )
            {
            itX.Set( (itD.Get())[0] );

            itY.Set( (itD.Get())[1] );
            }

            stringstream deltaXNames;
            deltaXNames << inputFilename << "_deltaX0" << j << ".tif";
            stringstream deltaYNames;
            deltaYNames << inputFilename << "_deltaY0" << j << ".tif";
            WriteFloatImage(deltaX, deltaXNames.str().c_str());
            WriteFloatImage(deltaY, deltaYNames.str().c_str());

            int listSize1 = xCoord1.size();
            int listSize2 = xCoord2.size();
            cout << "There are " << listSize1 << " nuclei in the fixed image and " <<
listSize2 << " in the moving image." << endl;
            FloatIndex current;
            FloatIndex modified;
            FloatIndex moving;

            int numMatches = 0;

            for(int n = 0; n < listSize1; n++)
            {

                  // current has the centroid coordinates in the fixed image
                  current[0] = xCoord1.at(n);
                  current[1] = yCoord1.at(n);
                  float xDisp = (deltaX->GetPixel(current));
                  float yDisp = (deltaY->GetPixel(current));

                  // modified has the centroid coordinates predicted in the moving
image using the registration filter
                  modified[0] = current[0] + xDisp;
                  modified[1] = current[1] + yDisp;

                  //match will count how many matches there are
                  //the vector matches will store all possible match coordinates
                  //the other two vectors will be used to determine the "best" match
                  int match = 0;
```

```
                        vector<int> matches;
                        vector<double> matchDistances;
                        vector<double> angleDiffs;
                        for(int m = 0; m < listSize2; m++)
                        {
                                // moving will contain all the centroids found with the binary
image filter
                                moving[0] = xCoord2.at(m);
                                moving[1] = yCoord2.at(m);

                                // distance finds the difference between coordinates found
with the binary filter and the registration filter
                                double distance = sqrt( ( (moving[0]-modified[0])*(moving[0]-
modified[0]) ) + ( (moving[1]-modified[1])*(moving[1]-modified[1])  )  );


                                if( distance <= maxDistance )  // if the distance is below a
certain threshold, we will store that as a "match" using the below
                                {
                                        match += 1; // counting number of possible matches
                                        matches.push_back(m);
                                        matchDistances.push_back(distance);
                                        double dispAngle =
GetAngle(current[0],current[1],modified[0],modified[1]);
                                        double matchAngle =
GetAngle(current[0],current[1],moving[0],moving[1]);
                                        double difference = abs(dispAngle-matchAngle);
                                        angleDiffs.push_back(difference);
                                }
                        }

                        int m;
                        int flag = 0;

                        if (match == 1) // only one match, so we choose that one
                        {
                                m = matches.at(0);
                        }
                        else if (match > 1) // multiple possible matches, so we will take
the minimum distance and choose that match
                        {
                                int distIndex = FindIndexOfMinimum(matchDistances);
                                int angleIndex = FindIndexOfMinimum(angleDiffs);
                                for (int u =0; u < matchDistances.size(); u++)
                                {
                                        //cout << "Match distance of nucleus " << matches.at(u)
<< ": " << matchDistances[u] << endl;
                                }
                                //cout << "Choosing nucleus: " << matches.at(distIndex) <<
endl;

                                if (distIndex == angleIndex)
                                {
                                        m = matches.at(distIndex); // taking the match that has
the least distance and angle differences
                                }
                                else
                                {
```

```
                                m = matches.at(distIndex); // need to come up with a
better criteria that considers angle too...
                        }
                }
                else if (match == 0)  // if there are no possible matches
                {
                        flag = 1; // the dummy variable flag tells us there are no
matches when it equals 1
                }

                if (flag == 0) // if there is a match present
                {

                        mapping1.push_back( n ); // mapping vectors will allow us to
map nucleus n to m since they will have the same index within the vectors
                        mapping2.push_back( m );

                                if (j == 0) // the first registration time
                                {
                                        global[numMatches].at(1) = n;
                                        global[numMatches].at(2) = m;
                                        unmatchedDisps[numMatches].at(1) =
xCoord2.at(m);
                                        unmatchedDisps[numMatches].at(2) =
yCoord2.at(m);
                                        mappingGL.push_back(global[numMatches].at(0));
                                }
                                else // any other registration
                                {
                                        int globalMatch = 0; // a dummy variable that
indicates if a global match was found

                                        for(int h = 0; h < global.size(); h++)
                                        {
                                                if (global[h].at(j+1) == n)
                                                {
                                                        global[h].at(j+2) = m;
                                                        unmatchedDisps[h].at(1) =
xCoord2.at(m);
                                                        unmatchedDisps[h].at(2) =
yCoord2.at(m);
                                                        //cout<<"Matching nucleus " << n
<< " to " << m << " and assigning global label " << global[h].at(0) << endl;

        mappingGL.push_back(global[h].at(0)); // global[h].at(0) will contain the global
label
                                                        globalMatch += 1;
                                                }
                                                if (globalMatch!= 0) // preventing more
than one match
                                                {
                                                        break;
                                                }

                                        }

                                        int index;
```

```
                                        if (globalMatch == 0) // if there were no global
matches
                                        {
                                                for(int h = 0; h < global.size(); h++)
                                                {
                                                        int flag1 = 0;

                                                        for (int q = 1; q < range; q++)
                                                        {
                                                                if ( (global[h].at(q) != -
1) ) // if the vector global[h] has been modified at all yet
                                                                {
                                                                        flag1 = 1;
        // if it has been modified, flag1 will equal 1
                                                                }
                                                        }

                                                        if (flag1 == 1) // if global[h]
has been modified
                                                        {
                                                                index = h+1; // the new
index will increase to the next vector until finally the first untouched vector is found
                                                        }
                                                }

                                                //cout << "final new index is " << index
<< "." << endl;

                                                //this will allow for a new globally
labeled vector
                                                mappingGL.push_back(global[index].at(0));

                                                global[index].at(j+1) = n;
                                                global[index].at(j+2) = m;
                                        }



                                }



                                float actualDistance = sqrt( ( (xCoord1.at(n)-
xCoord2.at(m))*(xCoord1.at(n)-xCoord2.at(m)) ) + ( (yCoord1.at(n)-
yCoord2.at(m))*(yCoord1.at(n)-yCoord2.at(m))  )  );
                                float thetaRadians = GetAngle( xCoord1.at(n),
yCoord1.at(n), xCoord2.at(m), yCoord2.at(m));
                                float thetaDeg = thetaRadians*(180/pi);
                                fprintf(pFile2,
"%i\t%i\t%i\t%i\t%i\t%i\t%i\t%i\t%f\t%f\t%f\n", j, n, m, mappingGL[numMatches],
xCoord1.at(n), yCoord1.at(n), xCoord2.at(m), yCoord2.at(m), actualDistance, thetaDeg,
thetaRadians);

                                numMatches += 1;

                        }

                }
```

```
		int numNucleiToMatch = listOfNucleiToMatch.size();
		int numNewlyFoundMatches = 0;

		cout << "There are " << numNucleiToMatch << " unmatched nuclei to try." <<
endl;

		for (int c = 0; c < numNucleiToMatch; c++)
		{
			int matchIndex = listOfNucleiToMatch.at(c);
			int matchGlobalIndex = unmatchedDisps[matchIndex].at(0);
			int predictedXCoord = Round(unmatchedDisps[matchIndex].at(3));
			int predictedYCoord = Round(unmatchedDisps[matchIndex].at(4));

			FloatIndex predicted;
			predicted[0] = predictedXCoord;
			predicted[1] = predictedYCoord;

			int newMatches = 0;
			vector<int> newMatchesList;
			vector<double> newMatchDistances;

			for(int d = 0; d < listSize2; d++)
			{
				int alreadyMatchedFlag = 0;

				for(int h = 0; h < global.size(); h++)
				{
					if (global[h].at(j+2) == d)
					{
						alreadyMatchedFlag = 1;
					}
				}

				if( alreadyMatchedFlag == 0)
				{
					// moving will contain all the centroids found with the
binary image filter
					moving[0] = xCoord2.at(d);
					moving[1] = yCoord2.at(d);

					double distance = sqrt( ( (moving[0]-
predicted[0])*(moving[0]-predicted[0]) ) + ( (moving[1]-predicted[1])*(moving[1]-
predicted[1])  )  );

					if( distance <= maxDistance )  // if the distance is
below a certain threshold, we will store that as a "match" using the below
					{
						cout << "Nuclei " << d << " is being matched to
" << matchGlobalIndex << "."<< endl;
						newMatches += 1; // counting number of possible
matches
						newMatchesList.push_back(d);
						newMatchDistances.push_back(distance);
					}
				}
			}
			int m;
```

```
                        int noMatchFlag = 0;
                        if (newMatches == 1) // only one match, so we choose that one
                        {
                                m = newMatchesList.at(0);
                        }

                        else if (newMatches > 1) // multiple possible matches, so we will
take the minimum distance and choose that match
                        {
                                int distIndex = FindIndexOfMinimum(newMatchDistances);
                                m = newMatchesList.at(distIndex); // taking the match that has
the least distance and angle differences
                        }
                        else if (newMatches == 0)  // if there are no possible matches
                        {
                                noMatchFlag = 1; // the dummy variable noMatchFlag tells us
there are no matches when it equals 1
                        }

                        if (noMatchFlag == 0) // if there is a match present
                        {
                                //first fill in the new position at global
                                global[matchIndex].at(j+2) = m;
                                //the modify unmatchedDisps so that we know the match has been
found
                                unmatchedDisps[matchIndex].at(1) = xCoord2.at(m);
                                unmatchedDisps[matchIndex].at(2) = yCoord2.at(m);
                                unmatchedDisps[matchIndex].at(3) = -1;
                                unmatchedDisps[matchIndex].at(4) = -1;
                                unmatchedDisps[matchIndex].at(5) = 0;
                                float actualDistance = sqrt( ( (predicted[0]-
xCoord2.at(m))*(predicted[0]-xCoord2.at(m)) ) + ( (predicted[1]-
yCoord2.at(m))*(predicted[1]-yCoord2.at(m))  )  );
                                float thetaRadians = GetAngle( predicted[0], predicted[1],
xCoord2.at(m), yCoord2.at(m));
                                float thetaDeg = thetaRadians*(180/pi);
                                fprintf(pFile2,
"%i\t%i\t%i\t%i\t%i\t%i\t%i\t%i\t%f\t%f\t%f\n", j, -1, m, matchGlobalIndex, predicted[0],
predicted[1], xCoord2.at(m), yCoord2.at(m), actualDistance, thetaDeg, thetaRadians);
                                numNewlyFoundMatches +=1;
                        }

                }
                cout << "The algorithm recovered " << numNewlyFoundMatches << " nuclei in
this frame." << endl;

                listOfNucleiToMatch.clear(); // this must be cleared in order to have a new
list of unmatched nuclei

                for (int k = 0; k < vectorSize; k++)
                {
                        if( unmatchedDisps[k].at(5) < unmatchedTimeLimit )
                        {
                                if ( (unmatchedDisps[k].at(1) != -1) && (global[k].at(j+2) ==
-1) ) // this will find nuclei that were mapped in the previous registration and not the
current one.
                                {
```

```
                                    cout << "Nucleus at global number " << global[k].at(0)
<< " was not matched for the "<< (unmatchedDisps[k].at(5)+1) << " time." << endl;
                                    FloatIndex mostRecentlyFound;
                                    double tempX;
                                    double tempY;

                                    if (unmatchedDisps[k].at(3) == -1) // this is the first
time it has not been matched
                                    {
                                           mostRecentlyFound[0] = unmatchedDisps[k].at(1);
                                           mostRecentlyFound[1] = unmatchedDisps[k].at(2);
                                           tempX = unmatchedDisps[k].at(1);
                                           tempY = unmatchedDisps[k].at(2);
                                           unmatchedDisps[k].at(5) += 1;
                                    }
                                    else // the nucleus has been left unmatched before
                                    {
                                           mostRecentlyFound[0] =
Round(unmatchedDisps[k].at(3));
                                           mostRecentlyFound[1] =
Round(unmatchedDisps[k].at(4));
                                           tempX = unmatchedDisps[k].at(3);
                                           tempY = unmatchedDisps[k].at(4);
                                           unmatchedDisps[k].at(5) += 1;
                                    }


                                    if ( (mostRecentlyFound[0]<=inputSize[0]) &&
(mostRecentlyFound[1]<=inputSize[1]) )
                                    {
                                           float xDisp = deltaX-
>GetPixel(mostRecentlyFound);
                                           float yDisp = deltaY-
>GetPixel(mostRecentlyFound);

                                           unmatchedDisps[k].at(3) = tempX + xDisp;
                                           unmatchedDisps[k].at(4) = tempY + yDisp;

                                           //cout << "Nucleus " << unmatchedDisps[k].at(0)
<< " is predicted to be at [" << unmatchedDisps[k].at(3) << ", " <<
unmatchedDisps[k].at(4) << "]." << endl;
                                           listOfNucleiToMatch.push_back(k); // this is
going to be saved to try and match these nuclei in the following frame
                                    }
                             }
                      }
              }


              RGBImageType::Pointer fixedImage = RGBImageType::New();
              fixedImage->SetRegions(deltaX->GetLargestPossibleRegion());
              fixedImage->Allocate();
              fixedImage = RGBextractor1->GetOutput();

              FloatIndex startingPoint;
              FloatIndex finalPoint;
              int mappingSize = mapping1.size();
```

```
		for (int i = 0; i < mappingSize; i++)
		{
			FloatIndex startingPoint;
			FloatIndex finalPoint;
			startingPoint[0] = xCoord1.at(mapping1[i]);
			startingPoint[1] = yCoord1.at(mapping1[i]);
			finalPoint[0] = xCoord2.at(mapping2[i]);
			finalPoint[1] = yCoord2.at(mapping2[i]);
			DrawDisplacement(fixedImage, startingPoint, finalPoint);
			RGBPixelType color;
			int colorStep = 255/range;
			color[0] = 255 - (j*colorStep);
			color[1] =  (j*colorStep);
			color[2] =  (j*colorStep);
			itk::LineIterator<RGBImageType> drawline(allDispImage,
startingPoint, finalPoint); drawline.GoToBegin();
				while (!drawline.IsAtEnd())  {drawline.Set(color);
++drawline;}
		}


		stringstream gridNames;
		gridNames << inputFilename << "_tracking0" << j << ".tif";

		RGBWriterType::Pointer RGBwriter = RGBWriterType::New();
		RGBwriter->SetFileName( gridNames.str().c_str() );
		RGBwriter->SetInput( fixedImage );

		TIFFIOType::Pointer tiffIO = TIFFIOType::New();
		//output image will be a .tif file
		RGBwriter->SetImageIO(tiffIO);
		RGBwriter->Update();

	}

	RGBWriterType::Pointer RGBwriter = RGBWriterType::New();
	stringstream dispNames;
	dispNames << inputFilename << "_AllDisplacements" << ".tif";
	RGBwriter->SetFileName( dispNames.str().c_str() );
	RGBwriter->SetInput( allDispImage );
	TIFFIOType::Pointer tiffIO = TIFFIOType::New();
	//output image will be a .tif file
	RGBwriter->SetImageIO(tiffIO);
	RGBwriter->Update();

	//deallocating memory
	delete [] inputFilename;


	system("PAUSE");
	return EXIT_SUCCESS;

}


//Functions~~~
```

```
int Round(float Number)
{
	return floor( Number + 0.5 );
}
//Round uses the generally accepted method of rounding floats to integers in C++, using
floor and adding 0.5 to simulate rounding.


int WriteImage( ImageType2D::Pointer ImageToWrite, const char* FileName ) {


	WriterType2D::Pointer writer = WriterType2D::New();
	writer->SetFileName( FileName );
	writer->SetInput( ImageToWrite );

	TIFFIOType::Pointer tiffIO = TIFFIOType::New();
	//output image will be a .tif file
	writer->SetImageIO(tiffIO);

	try
	{
		writer->Update();

    }
	catch (itk::ExceptionObject &e)
    {
		cerr << FileName << " could not be written." << endl << e << endl;   // if
there is a writing error, program will exit
		return EXIT_FAILURE;
	}
	return EXIT_SUCCESS;
}

int WriteShortImage( ShortImageType::Pointer ImageToWrite, const char* FileName ) {

	typedef itk::ImageFileWriter<ShortImageType> ShortWriter;
	ShortWriter::Pointer writer = ShortWriter::New();
	writer->SetFileName( FileName );
	writer->SetInput( ImageToWrite );

	TIFFIOType::Pointer tiffIO = TIFFIOType::New();
	//output image will be a .tif file
	writer->SetImageIO(tiffIO);

	try
	{
		writer->Update();

    }
	catch (itk::ExceptionObject &e)
    {
		cerr << FileName << " could not be written." << endl << e << endl;   // if
there is a writing error, program will exit
		return EXIT_FAILURE;
	}
	return EXIT_SUCCESS;
}
```

```
int WriteFloatImage( FloatImageType::Pointer ImageToWrite, const char* FileName ) {

	typedef itk::ImageFileWriter<FloatImageType> FloatWriter;
	FloatWriter::Pointer writer = FloatWriter::New();
	writer->SetFileName( FileName );
	writer->SetInput( ImageToWrite );

	TIFFIOType::Pointer tiffIO = TIFFIOType::New();
	//output image will be a .tif file
	writer->SetImageIO(tiffIO);

	try
	{
		writer->Update();

    }
	catch (itk::ExceptionObject &e)
    {
		cerr << FileName << " could not be written." << endl << e << endl;   // if
there is a writing error, program will exit
		return EXIT_FAILURE;
	}
	return EXIT_SUCCESS;
}
//WriteImage creates a new writer, takes the input file name, and writes the image to
file


void SetupIndices (int PointNumber, DispIndex* ArrayOfIndices, float*
ArrayOfIndicesActual, DispIndex Index, int XCoord, int YCoord )
{
	//assists to put the coordinates into an array of indices
	Index[0] = XCoord;
	Index[1] = YCoord;
	ArrayOfIndices[PointNumber] = Index;

	int actualPointNumber1 = (PointNumber*2);
	int actualPointNumber2 = (PointNumber*2)+1;
	ArrayOfIndicesActual[actualPointNumber1] = XCoord;
	ArrayOfIndicesActual[actualPointNumber2] = YCoord;
}

void UpdateIndices ( DispIndex* ArrayOfIndices, float* ArrayOfIndicesActual,
DispImage::Pointer DisplacementImage )
{
	for(int i=0; i<16; i++)
	{
		int n = i*2;
		DispIndex temp = ArrayOfIndices[i];
		float xDisp = (DisplacementImage->GetPixel(temp))[0];
		float yDisp = (DisplacementImage->GetPixel(temp))[1];
		ArrayOfIndicesActual[n] += xDisp;
		ArrayOfIndicesActual[n+1] += yDisp;
		((ArrayOfIndices[i])[0]) = Round(ArrayOfIndicesActual[n]);
		((ArrayOfIndices[i])[1]) = Round(ArrayOfIndicesActual[n+1]);
	}
```

```
	//for all 16 points stored in the ArrayOfIndices, the X- and Y-displacement values
are gotten using the GetPixel function.  Then those displacement values
	//are rounded and added to the original index value.
}

float GetAngle(int X1, int Y1, int X2, int Y2)
{
	float xDistance = X2 - X1;
	float yDistance = Y2 - Y1;
	float thetaRad;
	float pi = 3.14159265359;
	if(xDistance != 0 || yDistance != 0)
	{
		if(xDistance < 0)
		{
			thetaRad = atan(yDistance/xDistance) + pi/2;
		}
		else if (yDistance < 0)
		{
			thetaRad = atan(yDistance/xDistance) + pi;
		}
		else
		{
			thetaRad = atan(yDistance/xDistance);
		}
	}
	else
	{
		thetaRad = 0;
	}
	return thetaRad;
}


unsigned int GetCenterOfMass(unsigned int RegionLabel, unsigned int XorY,
LabelImageToLabelMapFilterType::Pointer filter)
{
	LabelImageToLabelMapFilterType::OutputImageType::LabelObjectType* labelObject =
filter->GetOutput()->GetNthLabelObject(RegionLabel);

	std::vector <unsigned int> coords;

	for(unsigned int pixelId = 0; pixelId < labelObject->Size(); pixelId++)
		{
	    coords.push_back(labelObject->GetIndex(pixelId)[XorY]);
		}

	unsigned int sumCoords = 0;
	for(std::vector<unsigned int>::iterator j = coords.begin(); j!=coords.end(); ++j)
	{sumCoords += *j;}

	unsigned int size = coords.size();
	unsigned int centerOfMass = sumCoords / size;

	return centerOfMass;
}
```

```
unsigned int GetSize(unsigned int RegionLabel, unsigned int XorY,
LabelImageToLabelMapFilterType::Pointer filter)
{
	LabelImageToLabelMapFilterType::OutputImageType::LabelObjectType* labelObject =
filter->GetOutput()->GetNthLabelObject(RegionLabel);

	std::vector <unsigned int> coords;

	for(unsigned int pixelId = 0; pixelId < labelObject->Size(); pixelId++)
		{
	    coords.push_back(labelObject->GetIndex(pixelId)[XorY]);
		}
	unsigned int size = coords.size();

	return size;
}

int DrawDisplacement ( RGBImageType::Pointer ImageToDrawOn, FloatIndex StartingPoint,
FloatIndex FinalPoint)
{
	float X;
	float Y;
	FloatIndex temp = StartingPoint;
	ImageType2D::SizeType size = ImageToDrawOn->GetLargestPossibleRegion().GetSize();
	FloatIndex terminalPoint = FinalPoint;
	if((temp !=
terminalPoint)&&(size[0]>terminalPoint[0])&&(size[1]>terminalPoint[1]))
	{
	FloatIndex arrowPoint1;
	FloatIndex arrowPoint2;
	float theta = 0.26179;  // 15 degrees
	float x1 = terminalPoint[0];
	float y1 = terminalPoint[1];
	float x2 = StartingPoint[0];
	float y2 = StartingPoint[1];
	float AB = sqrt(((x1-x2)*(x1-x2))+((y1-y2)*(y1-y2)));
	float v1 = (cos(theta)*(x2-x1) + sin(theta)*(y2-y1))/(AB);
	float v2 = (cos(theta)*(y2-y1) - sin(theta)*(x2-x1))/(AB);
	float v3 = (cos(-theta)*(x2-x1) + sin(-theta)*(y2-y1))/(AB);
	float v4 = (cos(-theta)*(y2-y1) - sin(-theta)*(x2-x1))/(AB);
	arrowPoint1[0] = Round(x1+5*v1);
	arrowPoint1[1] = Round(y1+5*v2);
	arrowPoint2[0] = Round(x1+5*v3);
	arrowPoint2[1] = Round(y1+5*v4);

	RGBPixelType green;
	green[0] = 0;
	green[1] = 255;
	green[2] = 0;

	itk::LineIterator<RGBImageType> drawline1(ImageToDrawOn, StartingPoint,
terminalPoint); drawline1.GoToBegin();
	while (!drawline1.IsAtEnd())  {drawline1.Set(green); ++drawline1;}

	if((size[0]>arrowPoint1[0])&&(size[1]>arrowPoint1[1]))
	{
		itk::LineIterator<RGBImageType> drawline2(ImageToDrawOn, terminalPoint,
arrowPoint1); drawline2.GoToBegin();
```

```
		while (!drawline2.IsAtEnd())  {drawline2.Set(green); ++drawline2;}
	}

	if((size[0]>arrowPoint2[0])&&(size[1]>arrowPoint2[1]))
	{
	itk::LineIterator<RGBImageType> drawline3(ImageToDrawOn, terminalPoint,
arrowPoint2); drawline3.GoToBegin();
	while (!drawline3.IsAtEnd())  {drawline3.Set(green); ++drawline3;}
	}

	}

	return EXIT_SUCCESS;
}

int FindIndexOfMinimum( vector<double> VectorToOperateOn )
{
	int size = VectorToOperateOn.size();
	double tempMin = VectorToOperateOn.at(0);
	for (int k = 0; k < size-1; k++)
	{
		double newMin = min(tempMin, VectorToOperateOn.at(k+1));
		tempMin = newMin;
	}

	int index;
	for (int k = 0; k < size; k++)
	{
		if (tempMin == VectorToOperateOn.at(k))
		{
			index = k;
		}
	}
	return index;
}

int CheckIfIndexIsInsideMask(
BinaryImageToLabelMapFilterType::OutputImageType::LabelObjectType*  LabelObject, int
xPoint, int yPoint )
{
	int flag = 0;

	for(unsigned int pixelId = 0; pixelId < LabelObject->Size(); pixelId++)
	{
		ImageType2D::IndexType index = LabelObject->GetIndex(pixelId);
		if(index[0]==xPoint && index[1]==yPoint)
		{
			flag = 1;
		}
	}

	return flag;
}

ShortImageType::Pointer PerformSegmentation(FloatImageType::Pointer anisoDiffImage,
FloatImageType::Pointer gradMagImage,
                                            const float threshold, const float level, const
float intensityThreshold,
```

```
                                        int maxNucSize, int minNucSize, float
minRoundness)
{
  typedef itk::WatershedImageFilter<FloatImageType> WatershedFilterType;
  WatershedFilterType::Pointer watershed = WatershedFilterType::New();
  watershed->SetThreshold(threshold);
  watershed->SetLevel(level);
  watershed->SetInput(gradMagImage);
  watershed->Update();

  typedef itk::ScalarToRGBColormapImageFilter<LabeledImageType, RGBImageType>
RGBFilterType;
  RGBFilterType::Pointer colormapImageFilter = RGBFilterType::New();
  colormapImageFilter->SetInput(watershed->GetOutput());
  colormapImageFilter->SetColormap( RGBFilterType::Jet );
  colormapImageFilter->Update();

  std::stringstream ss;
  ss << "output_" << threshold << "_" << level << "_colorMap.tif";
  std::stringstream ss1;
  ss1 << "output_" << threshold << "_" << level << "_labelImage.tif";
  std::stringstream ss2;
  ss2 << "output_" << threshold << "_" << level << "_labelImageExclusion.tif";
  std::stringstream ss3;
  ss3 << "output_" << threshold << "_" << level << "_labelImageExclusionMore.tif";

  typedef itk::ImageFileWriter<RGBImageType> FileWriterType;
  FileWriterType::Pointer writer = FileWriterType::New();
  writer->SetFileName(ss.str());
  writer->SetInput(colormapImageFilter->GetOutput());
  writer->Update();

  try{

        typedef itk::RescaleIntensityImageFilter< LabeledImageType, LabeledImageType >
RescaleFilterType;
        RescaleFilterType::Pointer rescaleFilter = RescaleFilterType::New();
        rescaleFilter->SetInput(watershed->GetOutput());
        rescaleFilter->SetOutputMinimum(0);
        rescaleFilter->SetOutputMaximum(65536);
        rescaleFilter->Update();

        typedef itk::CastImageFilter< LabeledImageType, ShortImageType > CastFilterType;
        CastFilterType::Pointer castFilter = CastFilterType::New();
        castFilter->SetInput(rescaleFilter->GetOutput());
        castFilter->Update();

        typedef itk::ImageFileWriter<ShortImageType> LabeledWriterType;
        LabeledWriterType::Pointer writer1 = LabeledWriterType::New();
        writer1->SetFileName(ss1.str());
        writer1->SetInput(castFilter->GetOutput());
        writer1->Update();

        // Remove label objects that have NUMBER_OF_PIXELS > maxNucleusSize and ROUNDNESS
< minRoundness
        typedef itk::LabelImageToShapeLabelMapFilter <ShortImageType>
LabelImageToLabelMapFilterType;
```

```
	LabelImageToLabelMapFilterType::Pointer labelImageToLabelMapFilter =
LabelImageToLabelMapFilterType::New ();
	labelImageToLabelMapFilter->SetInput(castFilter->GetOutput());
	labelImageToLabelMapFilter->Update();

	//the shape opening label map will allow us to implement the exclusion criteria
	typedef itk::ShapeOpeningLabelMapFilter<
LabelImageToLabelMapFilterType::OutputImageType > ShapeOpeningLabelMapFilterType;
	ShapeOpeningLabelMapFilterType::Pointer sizeShapeOpeningLabelMapFilter =
ShapeOpeningLabelMapFilterType::New();
	sizeShapeOpeningLabelMapFilter->SetInput( labelImageToLabelMapFilter->GetOutput()
);
	sizeShapeOpeningLabelMapFilter->SetLambda( maxNucSize );
	sizeShapeOpeningLabelMapFilter->ReverseOrderingOn();
	sizeShapeOpeningLabelMapFilter->SetAttribute(
ShapeOpeningLabelMapFilterType::LabelObjectType::NUMBER_OF_PIXELS);
	sizeShapeOpeningLabelMapFilter->Update();

	ShapeOpeningLabelMapFilterType::Pointer roundnessShapeOpeningLabelMapFilter =
ShapeOpeningLabelMapFilterType::New();
	roundnessShapeOpeningLabelMapFilter->SetInput( sizeShapeOpeningLabelMapFilter-
>GetOutput() );
	roundnessShapeOpeningLabelMapFilter->SetLambda( minRoundness );
	roundnessShapeOpeningLabelMapFilter->SetAttribute(
ShapeOpeningLabelMapFilterType::LabelObjectType::ROUNDNESS);
	roundnessShapeOpeningLabelMapFilter->Update();

	typedef
itk::LabelMapToLabelImageFilter<LabelImageToLabelMapFilterType::OutputImageType,
ShortImageType>	LabelMapToLabelImageFilterType;
	LabelMapToLabelImageFilterType::Pointer labelMapToLabelImageFilter =
LabelMapToLabelImageFilterType::New();
	labelMapToLabelImageFilter->SetInput(roundnessShapeOpeningLabelMapFilter-
>GetOutput());
	labelMapToLabelImageFilter->Update();

	writer1->SetFileName(ss2.str());
	writer1->SetInput(labelMapToLabelImageFilter->GetOutput());
	writer1->Update();

	typedef itk::LabelStatisticsImageFilter< FloatImageType, ShortImageType >
LabelStatisticsImageFilterType;
	LabelStatisticsImageFilterType::Pointer labelStatisticsImageFilter =
LabelStatisticsImageFilterType::New();
	labelStatisticsImageFilter->SetLabelInput( labelMapToLabelImageFilter->GetOutput()
);
	labelStatisticsImageFilter->SetInput(anisoDiffImage);
	labelStatisticsImageFilter->Update();

	//the iterator will operate on each nucleus at a time to find the centroid
	typedef itk::LabelGeometryImageFilter< ShortImageType >
	LabelGeometryImageFilterType;
	LabelGeometryImageFilterType::Pointer labelGeometryImageFilter =
LabelGeometryImageFilterType::New();
	labelGeometryImageFilter->SetInput( labelMapToLabelImageFilter->GetOutput() );
	labelGeometryImageFilter->Update();

	//creating an iterator will allow us to iterate over all of the nuclei regions
```

```
	LabelGeometryImageFilterType::LabelsType allLabels = labelGeometryImageFilter-
>GetLabels();
	LabelGeometryImageFilterType::LabelsType::iterator allLabelsIt;
	ShapeOpeningLabelMapFilterType::OutputImageType::Pointer tempMap =
roundnessShapeOpeningLabelMapFilter->GetOutput();

	for( allLabelsIt = allLabels.begin(); allLabelsIt != allLabels.end();
allLabelsIt++ )
	{
		LabelGeometryImageFilterType::LabelPixelType labelValue = *allLabelsIt;
		int label = labelValue;
		//ImageType2D::PointType centroid = labelGeometryImageFilter-
>GetCentroid(labelValue);
		//ImageType2D::IndexType centroidIndex;
		//centroidIndex[0]=centroid[0];
		//centroidIndex[1]=centroid[1];
		//int statsLabel = caster->GetOutput()->GetPixel(centroidIndex);

		if(label != 0)
		{
			float mean = labelStatisticsImageFilter->GetMean(labelValue);
			int size = labelGeometryImageFilter->GetVolume(labelValue);
			if(mean<intensityThreshold || size < minNucSize)
			{
				tempMap->RemoveLabel(label);
			}
		}
	}

	LabelMapToLabelImageFilterType::Pointer labelMapToLabelImageFilter2 =
LabelMapToLabelImageFilterType::New();
	labelMapToLabelImageFilter2->SetInput(tempMap);
	labelMapToLabelImageFilter2->Update();

	writer1->SetFileName(ss3.str());
	writer1->SetInput(labelMapToLabelImageFilter2->GetOutput());
	writer1->Update();

	ShortImageType::Pointer segmentedImage = labelMapToLabelImageFilter2->GetOutput();

	return segmentedImage;
  }

  catch( itk::ExceptionObject & err )
  {
    std::cerr << "ExceptionObject caught !" << std::endl;
    std::cerr << err << std::endl;
  }
}
```

B.1.4 Movement analysis code

```
(*Importing the data!!!  [[1]] necessary for importing excel format.  \
\\ necessary for formatting *)
(*JVMA Arguments are setting the amount of memory to allocate to the \
Java heap space.... "-Xmx___m" where ___ is the number of MB *)
(*for very large files with lots of nuclei, ~5000 is necessary *)
Needs["JLink`"];
ReinstallJava[JVMArguments -> "-Xmx5000m"];
data = N[Import[
    "C:\\Users\\Tim\\Desktop\\desktop\\cmbe data\\h2b \
tracking\\input.xlsx"][[1]]];
numData = Dimensions[data][[1]]
lengthData = Dimensions[data][[2]]
minReg = Min[data[[2 ;; numData, 1]]]
maxReg = Max[data[[2 ;; numData, 1]]]
minNum = Min[data[[2 ;; numData, 4]]]
maxNum = Max[data[[2 ;; numData, 4]]]

nuc1 = Cases[data, {_, _, _, 1000., _, _, _, _, _, _, _}];
nuc2 = Cases[data, {_, _, _, 1001., _, _, _, _, _, _, _}];
nuc1Coords = Join[{nuc1[[1, 5 ;; 6]]}, nuc1[[All, 7 ;; 8]]];
nuc2Coords = Join[{nuc2[[1, 5 ;; 6]]}, nuc2[[All, 7 ;; 8]]];
```

```
allNucCoords = {{}};
For[i = minNum, i < maxNum + 1, i++,
 (nuc = Cases[data, {_, _, _, N[i], _, _, _, _, _, _}]) &&
  (nucCoords = Join[{nuc[[1, 5 ;; 6]]}, nuc[[All, 7 ;; 8]]]) &&
  (allNucCoords = Join[allNucCoords, {nucCoords}])
 ]
allNucCoords = Drop[allNucCoords, 1];
scale = 775/512;
scaled = allNucCoords * scale;

firstPoints = allNucCoords[[All, 1, All]];
firstPointsScaled = firstPoints*scale;

plot1 = ListPlot[firstPointsScaled,
  PlotMarkers -> {Automatic, Medium}, PlotStyle -> Black]
plot2 = ListPlot[scaled, Joined -> True, AspectRatio -> 1,
  PlotRange -> {{0, 775}, {0, 775}}]
plot3 = ListPlot[scaled, Axes -> False, Joined -> True,
   AspectRatio -> 1, PlotRange -> {{0, 775}, {0, 775}}];
Show[plot2, plot1]
```

```
finalplot = Show[plot3, plot1];




image = Image[finalplot, ImageSize -> {512, 512}];
Export["C:\\Users\\Tim\\Desktop\\desktop\\cmbe data\\h2b \
tracking\\graph.tif", image];



numNuc = Dimensions[scaled][[1]]
Dimensions[scaled[[1]]][[1]]

minTrackedFrames = 50;
count = 0;
bestNucCoords = {{}};
For[k = 1, k < numNuc + 1, k++,
 (If[(Dimensions[scaled[[k]]][[1]] > minTrackedFrames),
   ((count++) && (bestNucCoords = Join[bestNucCoords, {scaled[[k]]}]))
   ])
```

```
 ]
count
bestNucCoords = Drop[bestNucCoords, 1];
Dimensions[bestNucCoords]


plot4 = ListPlot[bestNucCoords[[6]], Joined -> True, AspectRatio -> 1,
   PlotRange -> {{0, 775}, {0, 775}}]




netDisps = ConstantArray[0, {count}];
actualDisps = ConstantArray[0, {count}];
persistenceScores = ConstantArray[0, {count}];
Clear[p1, p2];
distance[p1_, p2_] =
  Sqrt[(p1[[1]] - p2[[1]])^2 + (p1[[2]] - p2[[2]])^2];


Part::partd : "Part specification \[NoBreak]p1\[LeftDoubleBracket]1\
\[RightDoubleBracket]\[NoBreak] is longer than depth of object. \
\!\(\*ButtonBox["\[RightSkeleton]",
Appearance->{Automatic, None},
BaseStyle->"Link",
ButtonData:>"paclet:ref/message/General/partd",
```

ButtonNote->"Part::partd"]\)"

Part::partd : "Part specification \[NoBreak]p2\[LeftDoubleBracket]1\
\[RightDoubleBracket]\[NoBreak] is longer than depth of object. \
\!\(*ButtonBox["\[RightSkeleton]",
Appearance->{Automatic, None},
BaseStyle->"Link",
ButtonData:>"paclet:ref/message/General/partd",
ButtonNote->"Part::partd"]\)"

Part::partd : "Part specification \[NoBreak]p1\[LeftDoubleBracket]2\
\[RightDoubleBracket]\[NoBreak] is longer than depth of object. \
\!\(*ButtonBox["\[RightSkeleton]",
Appearance->{Automatic, None},
BaseStyle->"Link",
ButtonData:>"paclet:ref/message/General/partd",
ButtonNote->"Part::partd"]\)"

General::stop : "Further output of \[NoBreak]\!\(*
StyleBox[
RowBox[{\"Part\", \"::\", \"partd\"}], \"MessageName\"]\)\[NoBreak] \
will be suppressed during this calculation. \!\(*ButtonBox["\
\[RightSkeleton]",

```
Appearance->{Automatic, None},
BaseStyle->"Link",
ButtonData:>"paclet:ref/message/General/stop",
ButtonNote->"General::stop"]\)"


time1 = 5;
time2 = 10;
netDispsOverTimeScores =
  ConstantArray[0, {count, IntegerPart[maxReg + 1]}];
actualDispsOverTimeScores =
  ConstantArray[0, {count, IntegerPart[maxReg + 1]}];
persistenceOverTimeScores =
  ConstantArray[0, {count, IntegerPart[maxReg + 1]}];
(*need to find some way to iterate over time and calculate \
persistence over time and put it in the giant matrix*)
For[c = 1, c < count + 1, c++,
  ((x1 = (bestNucCoords[[c]])[[
       time1]]) && (x2 = (bestNucCoords[[c]])[[time2]]) && (netDisps[[
       c]] = distance[x1, x2]))];
For[d = 1, d < count + 1, d++,
 ((coords = bestNucCoords[[d]]) && (sum = 0) && (numPoints =
     Dimensions[bestNucCoords[[d]]][[1]]) && (
    For[e = time1, e < (time2 + 1), e++,
```

```
    ((x1 = coords[[e]]) && (x2 = coords[[e + 1]]) && (sum +=
        distance[x1, x2]))]
    )
   && (actualDisps[[d]] = sum)
  )]
For[f = 1, f < count + 1, f++,
  persistenceScores[[f]] = (netDisps[[f]]/actualDisps[[f]])];


interval = 3;
For[t = 1, t < maxReg, t++,
  ((t1 = t) && (t2 = t + interval)) && (
    For[c = 1, c < count, c++,
     (numTimePts = Dimensions[bestNucCoords[[c]]][[1]]) &&
      (If[t2 < numTimePts,
        ((x1 = (bestNucCoords[[c]])[[
            t1]]) && (x2 = (bestNucCoords[[c]])[[
            t2]]) && (netDispsOverTimeScores[[c, t]] =
          distance[x1, x2]))])]) &&
   (For[d = 1, d < count, d++,
     (numTimePts = Dimensions[bestNucCoords[[d]]][[1]]) &&
      (If[t2 < numTimePts,
        ((coords = bestNucCoords[[d]]) && (sum = 0) && (numPoints =
          Dimensions[bestNucCoords[[d]]][[1]]) && (
```

```
          For[e = t1, e < (t2 + 1), e++,
           ((x1 = coords[[e]]) && (x2 = coords[[e + 1]]) && (sum +=
              distance[x1, x2]))]
           )
          && (actualDispsOverTimeScores[[d, t]] = sum)
         )])]) &&
   (For[f = 1, f < count, f++,
     (numTimePts = Dimensions[bestNucCoords[[f]]][[1]]) && (If[
       t < numTimePts, (persistenceOverTimeScores[[f,
         t]] = (netDispsOverTimeScores[[f, t]]/
          actualDispsOverTimeScores[[f, t]]))])
     ])];
persistenceScores;
Histogram[persistenceScores, {0.1},
 ChartElementFunction -> "GradientScaleRectangle"]



vector[p1_, p2_] = {p2[[1]] - p1[[1]], p2[[2]] - p1[[2]]};
correlation[v1_, v2_] = (v1.v2)/(Norm[v1]*Norm[v2]);


Part::partd : "Part specification \[NoBreak]p2\[LeftDoubleBracket]1\
\[RightDoubleBracket]\[NoBreak] is longer than depth of object. \
\!\(\*ButtonBox["\[RightSkeleton]",
```

Appearance->{Automatic, None},

BaseStyle->"Link",

ButtonData:>"paclet:ref/message/General/partd",

ButtonNote->"Part::partd"]\)"

Part::partd : "Part specification \[NoBreak]p1\[LeftDoubleBracket]1\

\[RightDoubleBracket]\[NoBreak] is longer than depth of object. \

\!\(*ButtonBox["\[RightSkeleton]",

Appearance->{Automatic, None},

BaseStyle->"Link",

ButtonData:>"paclet:ref/message/General/partd",

ButtonNote->"Part::partd"]\)"

Part::partd : "Part specification \[NoBreak]p2\[LeftDoubleBracket]2\

\[RightDoubleBracket]\[NoBreak] is longer than depth of object. \

\!\(*ButtonBox["\[RightSkeleton]",

Appearance->{Automatic, None},

BaseStyle->"Link",

ButtonData:>"paclet:ref/message/General/partd",

ButtonNote->"Part::partd"]\)"

General::stop : "Further output of \[NoBreak]\!\(*

StyleBox[

```
RowBox[{\"Part\", \"::\", \"partd\"}], \"MessageName\"]\)\[NoBreak] \
will be suppressed during this calculation. \!\(\*ButtonBox["\
\[RightSkeleton]",
Appearance->{Automatic, None},
BaseStyle->"Link",
ButtonData:>"paclet:ref/message/General/stop",
ButtonNote->"General::stop"]\)"

distance[bestNucCoords[[1]][[1]], bestNucCoords[[17]][[1]]];

corrTimeScores = ConstantArray[0, {count, IntegerPart[maxReg + 1]}];
maxDistance = 50;
testInterval = 1;
corrScores = ConstantArray[0, {count}];
For[g = 1, g < count + 1, g++,
 ((maxTime = Dimensions[bestNucCoords[[g]]][[1]]) && (refNuc =
    bestNucCoords[[g]]) && (corrSumOverT = 0) && (corrCountOverT =
    0) &&
   (For[t = 1, t < (maxTime + 1 - testInterval), t++,
    ((refPoint = refNuc[[t]]) && (refPoint2 =
       refNuc[[t + testInterval]]) && (corrSum = 0) && (corrCount =
        0) &&
      (For[h = 1, h < count + 1, h++,
```

```
((testMaxTime = Dimensions[bestNucCoords[[h]]][[1]])
 If[((h != g) && (t + testInterval <= testMaxTime)),
  ((testPoint =
     bestNucCoords[[h]][[t]]) && (testPoint2 =
      bestNucCoords[[h]][[t + testInterval]])
    (If[distance[refPoint, testPoint] < maxDistance,
     ((V1 = vector[refPoint2, refPoint]) && (V2 =
       vector[testPoint2,


       testPoint]) && (If[(Norm[V1] != 0 &&
       Norm[V2 ] !=
       0), ((corrSum +=
       correlation[V1, V2]) && (corrCount++))]))
     ])
  )])]) &&
(If[
 corrCount !=
  0, (avgCorr =
   corrSum/corrCount) && (corrTimeScores[[g, t]] =
   avgCorr) && (corrSumOverT +=
   avgCorr) && (corrCountOverT++)]))]) &&
(If[corrCountOverT != 0,
 corrScores[[g]] = (corrSumOverT/corrCountOverT)]))];
```

```
Histogram[corrScores, {0.05},
 ChartElementFunction -> "GradientScaleRectangle"]



tempMax = Dimensions[bestNucCoords][[1]];
timeScores = ConstantArray[0, {count}];
For[i = 1, i < tempMax + 1, i++,
 timeScores[[i]] = Dimensions[bestNucCoords[[i]]][[1]]
 ]
Histogram[timeScores, {5},
 ChartElementFunction -> "GradientScaleRectangle"]



maxDistance = 50;
testInterval = 2;
distScores = ConstantArray[0, {count}];
For[g = 1, g < count + 1, g++,
  ((maxTime = Dimensions[bestNucCoords[[g]]][[1]]) && (refNuc =
     bestNucCoords[[g]]) && (corrSumOverT = 0) && (corrCountOverT =
     0) &&
```

```
(For[t = 1, t < (maxTime + 1 - testInterval), t++,
  ((refPoint = refNuc[[t]]) && (refPoint2 =
     refNuc[[t + testInterval]]) && (corrSum = 0) && (corrCount =
     0) &&
    (For[h = 1, h < count + 1, h++,
     ((testMaxTime = Dimensions[bestNucCoords[[h]]][[1]])
       If[((h != g) && (t + testInterval <= testMaxTime)),
        ((testPoint =
           bestNucCoords[[h]][[t]]) && (testPoint2 =
            bestNucCoords[[h]][[t + testInterval]])
          (If[distance[refPoint, testPoint] < maxDistance,
           ((D1 = distance[refPoint2, refPoint]) && (D2 =
             distance[testPoint2,
             testPoint]) && (If[(D1 != 0 &&
             D2 != 0), ((corrSum += (D1/
             D2)) && (corrCount++))]))
           ])
         )])]) &&
    (If[
     corrCount !=
      0, (avgCorr = corrSum/corrCount) && (corrSumOverT +=
       avgCorr) && (corrCountOverT++)]))]) &&
(If[corrCountOverT != 0,
```

```
    distScores[[g]] = (corrSumOverT/corrCountOverT)]))];


Histogram[Abs[distScores - 1], {0.05},
 ChartElementFunction -> "GradientScaleRectangle"]


tempMax2 = Dimensions[scaled][[1]];
timeScores2 = ConstantArray[0, {tempMax2}];
For[i = 1, i < tempMax2 + 1, i++,
 timeScores2[[i]] = Dimensions[scaled[[i]]][[1]]
 ]

Histogram[timeScores2, {5},
 ChartElementFunction -> "GradientScaleRectangle"]


time = N[(Array[FromDigits[{#}] &, {75}])*10/60]


ListPlot[{Reverse[Select[corrTimeScores[[99]], # != 0 &]],
```

```
 Reverse[Select[corrTimeScores[[29]], # != 0 &]]}, Joined -> True,
 PlotRange -> {{0, 75}, {-1, 1}}]

time // TableForm

corrTimeScores[[90]]
persistenceOverTimeScores[[10]]
persistenceOverTimeScores[[60]][[99]]
persistenceOverTimeScores[[60, 99]]
Export["C:\\Users\\Tim\\Desktop\\desktop\\cmbe data\\h2b \
tracking\\outputPersistence.xls", persistenceOverTimeScores]
Export["C:\\Users\\Tim\\Desktop\\desktop\\cmbe data\\h2b \
tracking\\outputCorrelation.xls", corrTimeScores]
```

B.2 LOW-RESOLUTION TRACTION FORCE MICROSCOPY

The following protocol outlines the process of gel preparation, *ex vivo* tissue isolate preparation, imaging and analysis for the low-resolution traction force microscopy experiments.

B.2.1 Gel preparation protocol

0.) Clean glass using the normal procedure. You will need:

a. One clean 45x50-1.5 glass coverslip used to image through

b. One clean 22x22 glass coverslip cut into thirds and then into fourths (roughly 3.5x11 each) to shape the gel.

1.) Treat the 45x50 glass coverslip with APES for 4-5min.

a. Using a micropipette, put approximately 300 uL on the glass and use the micropipette tip to spread the APES (aminopropyltrimethoxysilane) around, avoiding the edge of the glass.

b. When finished, **thoroughly** rinse the glass, using the ddH_2O squirt bottle. Leftover APES will react with glutaraldehyde in the next step if not careful.

2.) Put the glass coverslip in a 0.5% glutaraldehyde bath.

a. Prepare 10 mL of the bath per coverslip, combining 71.4 uL glutaraldehyde in 10 mL of pH 8 MBS.
b. Place the coverslip in the bath and let sit for 30 minutes.
c. Thoroughly rinse the glutaraldehyde off, and place the glass coverslip on a drying rack. Placing it in the 37C incubator will speed up the drying process. Be careful to keep track of which side of the glass has been silanized!
d. Silanized glass can be stored in a vacuum chamber for up to 2 weeks.

3.) Mix together acrylamide, bis-acrylamide, MBS, fibronectin, and beads and de-gas.

a. For a 0.05% polyacrylamide solution, use the following quantities:
    i. 42 uL of 30% acrylamide
    ii. 12.5 uL of 1% bis-acrylamide
    iii. 118 uL of pH 8 1/3X MBS
    iv. 25 uL of fibronectin (or 3 uL of rhodamine dextran + 22 uL of pH 8 MBS, if doing a control gel)
    v. 10 uL of the dark red 0.04 um beads (or 10 uL of pH 8 MBS, if doing a control)
b. Vortex solution/pipette up and down
c. Place in vacuum container and seal to de-gas.

4.) Make TEMED, APS, and finally, the NHS solution and QUICKLY add to the mixture.

a. For TEMED: combine 985 uL of pH 8 MBS with 15 uL of TEMED to get a 1.5% solution.
b. For APS (ammonium persulfate): combine 1 mL of pH 8 MBS with 0.025 g of APS to get a 2.5% solution.
c. Obtain the dry, silanized 45x50 glass coverslip and your de-gassed polyacrylamide cocktail.
d. The following should all be done VERY quickly:
    i. Measure out 0.015 g of acrylic acid NHS (N-hydroxysuccinimide). *When everything else is ready*, add NHS to 1 mL of high-quality DMSO to get a 15 mg/mL solution. Vortex until NHS dissolves and ***immediately*** add 2.5 uL NHS to the 30 uL of the pH 8 MBS, mix by pipetting, then add that entire 32.5 uL mixture to the polyacrylamide cocktail.
    ii. ***Immediately*** add 30 uL of 1.5% TEMED and 10 uL of 2.5% APS. Pipette solution up and down to mix evenly, while being careful not to create air bubbles.
    iii. For a 100 um gel, pipette 15.4 uL of the solution on the silanized 45x50 glass coverslip. Carefully place a 7x22 glass coverslip on top.
    iv. Flip larger coverslip to let gel hang upside-down, which will permit more beads to be at the surface of the polymerized gel. Place gel in a humid chamber.

5.) **Allow about 10 minutes for polymerization. After 10 minutes, flip gel rightside-up, frame with a PDMS chamber sealed with vacuum grease, and add normal pH MBS. Allow polymerization to continue for at least another 30 minutes.**

6.) **Prepare gel for imaging:**

a. Add regular 1/3 X MBS to the dish and using forceps under a stereoscope, remove the overlying 3.5x11 glass coverslip.

b. **At this point, you may want to check your gel for abnormalities using the Leica SP5 microscope. XZY mode is very useful for detecting uneven polymerization.**

c. Thoroughly rinse gel with MBS or DFA, depending on how it will be used.

d. Fill chamber with media and you are now ready to put an explant on it.

Table 3. Quick reference table for traction force gel recipe

	Chemical	Stock solution preparation/storage	Amount added for 0.05% polyacrylamide gels
First part	**Acrylamide (30%)**	3 g acrylamide in ddH2O, increase volume to 10 mL Stores "indefinitely" in 4C fridge, cover with foil	42 uL
	Bis-acrylamide (1%)	0.1 g bis-acrylamide in ddH2O, increase volume to 10 mL Stores "indefinitely" in 4C fridge, cover with foil	12.5 uL
	pH 8 MBS	N/A	88 uL
	Fibronectin	N/A	25 uL
	Bead solution	1 uL of 0.04 um dark-red beads in 29 uL pH 8 MBS Stores for about one week in 4C fridge, cover with foil	10 uL
Second part	**Acrylic acid NHS (15 mg/mL)**	0.015 g in 1 mL high-quality DMSO CANNOT store – must be made and used **immediately**	2.5 uL + 30 uL MBS
	TEMED (1.5%)	30 uL TEMED in 970 uL Should not store – make fresh every time	30 uL
	APS (2.5%)	0.050 g in 1000 uL pH 8 MBS Should not store – make fresh every time	10 uL

B.2.2 Explant preparation protocol

1.) Inject a membrane label into 1-cell stage embryos. I most often chose mem-GFP or mem-mKO2 because I most often used a combination of red and far red beads in the gel.

2.) Prepare 1.5% agarose plate for microsurgery and DFA supplemented with antibiotic. DO NOT USE BSA – it will inhibit explant binding to the gel.

3.) Use the forceps to dig a well to stabilize the embryo.

4.) Cut off the superficial ectoderm in the ventral region, extending all the way to the anterior lateral plate mesoderm.

5.) Once the ectoderm is off, slowly cut away the rest of the embryo to isolate the ventral mesoderm/endoderm. I found the following process to create the most consistent explants (Fig. 48A).

a. Cut the embryo in half along a transverse plane to remove the posterior half of the embryo.

b. Cut from the anterior "top" of the exposed mesoderm region along the neural tube. This will enable you to flatten the tissue, forming an arch shape.

c. Use the hair loop to remove everything except the exposed lateral and ventral mesoderm, along with its underlying endoderm.

6.) Mount the explant onto the traction force gel. Gently compress with a coverslip, making sure the coverslip is long enough to span the gel and silicone grease can reach the glass on either side (Fig. 48B).

7.) Image the explant for the experiment, making sure to minimize possibility of phototoxicity. At the beginning of the experiment, stabilize chamber as much as possible with clay so that when you trypsinize the explant in the next step, the chamber does not move considerably.

8.) At the conclusion of time lapse, gently wash out the media and replace it with trypsin to get the explants to release from the gel.

9.) Capture a "zero-force" image of the gel, i.e. without the explant adhered.

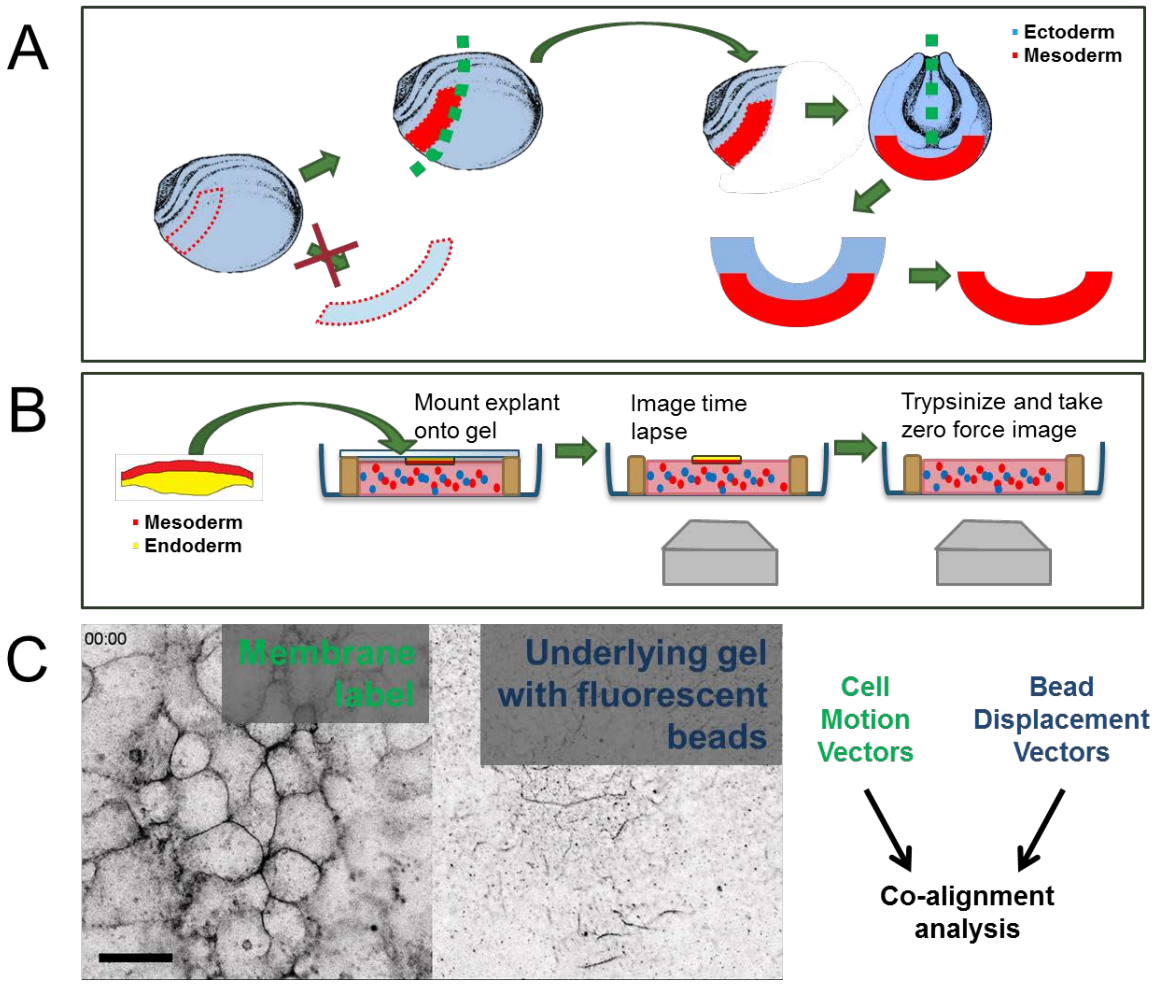

Figure 48. Schematics of microsurgery, imaging and analysis for low-resolution traction force microscopy

(A) Process of microsurgically isolating heart forming region. See text for details. (B) Process of imaging explant on traction force reporting gel. (C) Example of TFM experiment with HPCs on left and beads on right, which are used to calculate cell motion vectors and bead displacement vectors to feed into co-alignment analysis.

B.2.3 Image analysis protocol

1.) First, you need to perform a rigid registration between the zeroForce images and the time lapse. The ImageJ StackReg plugin was effective, but could only register one bead channel at a time. Consequentially, this method was not applicable for the membrane channel, which has nothing in its zeroForce image to register. In order to concurrently register the bead channels and the membrane channel, a custom rigid registration algorithm was implemented in ITK (Section B.2.4). This algorithm operates on an ROI set by the user, with the intention of that ROI containing a portion of the gel that does NOT have the explant over top of it (thus registration would not be affected by cell induced bead displacements).

2.) Once registered, each of the time lapses with their corresponding zeroForce images can be loaded into the TFMAnalysis program, also written with ITK (Section B.2.5). The algorithm uses FEMregistration filters to calculate bead and cell displacements, which are then printed out for each frame. Cell displacements are relative to the previous frame, while bead displacements are relative to the zeroForce image.

3.) Once all of the displacements are printed out, I used ImageJ to average the displacements over a 30min period using "Grouped Z Project." Averaging removed the displacement oscillations due to protrusion/retraction cycles and instead focused on larger trends. Then I used ImageJ's "Image Calculator" to calculate the cosine similarity score between the bead displacements (B) and cell displacements (C) using the following formula:

$$similarity = cos(\theta) = \frac{B \cdot C}{\|B\|\|C\|}$$

B.2.4 Rigid registration code

```
//including all necessary headers:
#include "itkImage.h"
#include <iostream>
#include <stdio.h>
#include <string>
#include <math.h>
#include "itkImageRegionConstIterator.h"
#include "itkImageRegionIterator.h"
#include "itkImageRegionIteratorWithIndex.h"
#include "itkImageFileReader.h"
#include "itkImageFileWriter.h"
#include "itkTIFFImageIO.h"
#include "itkLineIterator.h"
#include "itkImageSeriesWriter.h"

#include "itkRescaleIntensityImageFilter.h"
#include "itkHistogramMatchingImageFilter.h"
#include "itkFEMRegistrationFilter.h"
#include "itkExtractImageFilter.h"
#include "itkMaskImageFilter.h"
#include "itkConnectedThresholdImageFilter.h"
#include "itkCurvatureFlowImageFilter.h"
#include "itkCastImageFilter.h"
#include <itkRGBPixel.h>
#include "itkBlobSpatialObject.h"
#include "itkSpatialObjectPoint.h"

#include "itkRegionOfInterestImageFilter.h"
#include "itkImageRegistrationMethod.h"
#include "itkLinearInterpolateImageFunction.h"
#include "itkMeanSquaresImageToImageMetric.h"
#include "itkRegularStepGradientDescentOptimizer.h"
#include "itkResampleImageFilter.h"
#include "itkSpatialObjectToImageFilter.h"
#include "itkTranslationTransform.h"


using namespace std;
using namespace itk;

//function
prototypes~~~
```

```
	typedef ImageFileReader<CharImageType> ReaderType;
	typedef TIFFImageIO TIFFIOType;
	//First I will make a variable for the input filename, which will be put in at
command line using the cin function
	char* inputFilename = new char[50];

	cout << "Please enter the stack's file name (must be a TIFF file)." << endl;
	cout << "WARNING:: Entering more than 50 characters will cause an error." << endl;
	cout << "Input stack file name: ";
	cin >> inputFilename;

	stringstream memNames;
	memNames << inputFilename << "_mem.tif" ;
	stringstream beadNames;
	beadNames << inputFilename << "_redBeads.tif" ;
	stringstream fixedBeadNames;
	fixedBeadNames << inputFilename << "_redBeads_zeroForce.tif" ;

	cout<< memNames.str() << endl;

	TIFFIOType::Pointer tiffIO = TIFFIOType::New();

	Reader3DType::Pointer beadReader = Reader3DType::New();
			//creating the reader
    beadReader->SetFileName( beadNames.str().c_str() );
	//beadReader->SetImageIO(tiffIO);
	//the reader will now look for the file name input by the user
	Reader3DType::Pointer memReader = Reader3DType::New();
			//creating the reader
    memReader->SetFileName( memNames.str().c_str() );
	//memReader->SetImageIO(tiffIO);
	ReaderType::Pointer zeroForceReader = ReaderType::New();
	zeroForceReader->SetFileName( fixedBeadNames.str().c_str() );
	//zeroForceReader->SetImageIO(tiffIO);

	try
    {
		cout<< "1..."<<endl;
		beadReader->Update();
		cout<< "2..."<<endl;
		memReader->Update();
		cout<< "3..."<<endl;
		zeroForceReader->Update();// reader will attempt to update with inputted
filename
    }
	catch (itk::ExceptionObject &e)
    {
    cerr << "Image not found." << endl << e << endl;
			// if file cannot be found, program will exit
    delete inputFilename;
	return EXIT_FAILURE;
    }

	cout << "File found." << endl;
	float xFixed = 73.250;
	float yFixed = 969.750;
                                                //making a variable for the x and y
coords in fixed image
```

```
	//cout << "Please enter coordinates of first landmark in zero force image." <<
endl;
	//cout << "X: ";
	//cin >> xFixed;
	//cout << "Y: ";
	//cin >> yFixed;

	float xMoving = 71.625 ;
	float yMoving = 967.750;
                                                    //making a variable for the x and y
coords in fixed image
	/*cout << "Please enter coordinates of first landmark in moving image." << endl;
	cout << "X: ";
	cin >> xMoving;
	cout << "Y: ";
	cin >> yMoving;*/

	float xFixed2 = 985.750 ;
	float yFixed2 = 85;
                                           //making a variable for the x and y coords in
fixed image
	/*cout << "Please enter coordinates of second landmark in zero force image." <<
endl;
	cout << "X: ";
	cin >> xFixed2;
	cout << "Y: ";
	cin >> yFixed2;*/

	float xMoving2 = 983.333;
	float yMoving2 = 83.167 ;
                                                    //making a variable for the x and y
coords in fixed image
	/*cout << "Please enter coordinates of second landmark in moving image." << endl;
	cout << "X: ";
	cin >> xMoving2;
	cout << "Y: ";
	cin >> yMoving2;*/

	float xFixed3 = 975;
	float yFixed3 = 352.250;
                                                    //making a variable for the x and y
coords in fixed image
	/*cout << "Please enter coordinates of third landmark in zero force image." <<
endl;
	cout << "X: ";
	cin >> xFixed3;
	cout << "Y: ";
	cin >> yFixed3;*/

	float xMoving3 = 973;
	float yMoving3 = 351.50;


	//establishing variables for the input image's largest possible region and size
	Char3DImageType::RegionType inputRegion = (beadReader->GetOutput())-
>GetLargestPossibleRegion();
	Char3DImageType::SizeType inputSize = (beadReader->GetOutput())-
>GetLargestPossibleRegion().GetSize();
```

```
	int numFrames = inputSize[2];
	Char3DImageType::SizeType desiredSize = inputSize;
	desiredSize[2]=0;

	int radius = 30;
	CharImageType::IndexType roiStart1;
	CharImageType::IndexType roiStart2;
	CharImageType::IndexType roiStart3;
	roiStart1[0]= Round(xFixed-radius);		roiStart1[1]= Round(yFixed-
radius);
	roiStart2[0]= Round(xFixed2-radius);	roiStart2[1]= Round(yFixed2-radius);
	roiStart3[0]= Round(xFixed3-radius);	roiStart3[1]= Round(yFixed3-radius);

	CharImageType::SizeType roiSize;
	roiSize[0]=2*radius+1;
	roiSize[1]=2*radius+1;

	CharImageType::RegionType roiRegion1;
	CharImageType::RegionType roiRegion2;
	CharImageType::RegionType roiRegion3;
	roiRegion1.SetSize(roiSize);	roiRegion1.SetIndex(roiStart1);
	roiRegion2.SetSize(roiSize);	roiRegion2.SetIndex(roiStart2);
	roiRegion3.SetSize(roiSize);	roiRegion3.SetIndex(roiStart3);

	// Finally the for loop, which will for iterative registrations, is setup.  The
range input by the user is used as the stopping point here.
	for(unsigned int t = 0; t < numFrames; t++)
	{
		// ExtractFilterType will allow for the extraction of 2D slices from the 3D
stack read in by the reader.  Two extractors are necessary
		// because the registration filter requires two images for a registration.
		// Initializing them now makes sense because these will be constant
throughout the registration loop.

		typedef ExtractImageFilter<Char3DImageType,CharImageType>
ExtractFilterType;
		ExtractFilterType::Pointer memExtractor = ExtractFilterType::New();
		memExtractor->InPlaceOn();
		memExtractor->SetDirectionCollapseToSubmatrix();

		ExtractFilterType::Pointer beadExtractor = ExtractFilterType::New();
		beadExtractor->InPlaceOn();
		beadExtractor->SetDirectionCollapseToSubmatrix();

		//Extractor1 will get slice j while extractor2 gets slice (j+1).  These can
then be plugged into the registration filter.
		Char3DImageType::IndexType extractorStart = beadReader->GetOutput()-
>GetLargestPossibleRegion().GetIndex();
		extractorStart[2] = t;
		Char3DImageType::RegionType desiredRegion;
		desiredRegion.SetSize( desiredSize );
		desiredRegion.SetIndex( extractorStart );
		beadExtractor->SetExtractionRegion( desiredRegion );
		beadExtractor->SetInput( beadReader->GetOutput() );
		beadExtractor->Update();
		memExtractor->SetExtractionRegion( desiredRegion );
		memExtractor->SetInput( memReader->GetOutput() );
		//memExtractor->Update();
```

```
            cout<<"Casting..."<<endl;
            typedef itk::CastImageFilter<CharImageType, FloatImageType> CastFloatType;
            CastFloatType::Pointer castFloat = CastFloatType::New();
            castFloat->SetInput(beadExtractor->GetOutput());
            castFloat->Update();

            CastFloatType::Pointer castZeroForce = CastFloatType::New();
            castZeroForce->SetInput(zeroForceReader->GetOutput());
            castZeroForce->Update();

            cout<<"Histogram Matching..."<<endl;
            typedef itk::HistogramMatchingImageFilter<FloatImageType, FloatImageType>
HEFilterType;
            HEFilterType::Pointer IntensityEqualizeFilter = HEFilterType::New();
            IntensityEqualizeFilter->SetReferenceImage( castZeroForce->GetOutput() );
            IntensityEqualizeFilter->SetInput( castFloat->GetOutput() );
            IntensityEqualizeFilter->SetNumberOfHistogramLevels(150);
            IntensityEqualizeFilter->SetNumberOfMatchPoints(15);
            IntensityEqualizeFilter->ThresholdAtMeanIntensityOn();
            IntensityEqualizeFilter->Update();

            typedef itk::CastImageFilter<FloatImageType, CharImageType> CastCharType;
            CastCharType::Pointer caster = CastCharType::New();
            caster->SetInput(IntensityEqualizeFilter->GetOutput());
            caster->Update();

            cout<<"Defining ROIs..."<<endl;
            typedef RegionOfInterestImageFilter<CharImageType, CharImageType>
ROIFilterType;
            ROIFilterType::Pointer fixedROI1 = ROIFilterType::New();
            ROIFilterType::Pointer fixedROI2 = ROIFilterType::New();
            ROIFilterType::Pointer fixedROI3 = ROIFilterType::New();
            ROIFilterType::Pointer movingROI1 = ROIFilterType::New();
            ROIFilterType::Pointer movingROI2 = ROIFilterType::New();
            ROIFilterType::Pointer movingROI3 = ROIFilterType::New();

            fixedROI1->SetRegionOfInterest(roiRegion1);                  fixedROI1-
>SetInput(zeroForceReader->GetOutput());                     fixedROI1->Update();
            fixedROI2->SetRegionOfInterest(roiRegion2);                  fixedROI2-
>SetInput(zeroForceReader->GetOutput());                     fixedROI2->Update();
            fixedROI3->SetRegionOfInterest(roiRegion3);                  fixedROI3-
>SetInput(zeroForceReader->GetOutput());                     fixedROI3->Update();
            movingROI1->SetRegionOfInterest(roiRegion1);           movingROI1-
>SetInput(caster->GetOutput());                           movingROI1->Update();
            movingROI2->SetRegionOfInterest(roiRegion2);           movingROI2-
>SetInput(caster->GetOutput());                           movingROI2->Update();
            movingROI3->SetRegionOfInterest(roiRegion3);           movingROI3-
>SetInput(caster->GetOutput());                           movingROI3->Update();

            //  The transform that will map the fixed image into the moving image.
            typedef itk::TranslationTransform< double, 2 >
                  TransformType;
            typedef itk::RegularStepGradientDescentOptimizer
            OptimizerType;
            typedef itk::MeanSquaresImageToImageMetric< CharImageType, CharImageType >
MetricType;
```

```
            typedef itk:: LinearInterpolateImageFunction<CharImageType, double>
      InterpolatorType;
            typedef itk::ImageRegistrationMethod<CharImageType, CharImageType >
      RegistrationType;

            // Create components
            MetricType::Pointer         metric1        = MetricType::New();
            TransformType::Pointer      transform1     = TransformType::New();
            OptimizerType::Pointer      optimizer1     = OptimizerType::New();
            InterpolatorType::Pointer   interpolator1  = InterpolatorType::New();
            RegistrationType::Pointer   registration1  = RegistrationType::New();

            MetricType::Pointer         metric2        = MetricType::New();
            TransformType::Pointer      transform2     = TransformType::New();
            OptimizerType::Pointer      optimizer2     = OptimizerType::New();
            InterpolatorType::Pointer   interpolator2  = InterpolatorType::New();
            RegistrationType::Pointer   registration2  = RegistrationType::New();

            MetricType::Pointer         metric3        = MetricType::New();
            TransformType::Pointer      transform3     = TransformType::New();
            OptimizerType::Pointer      optimizer3     = OptimizerType::New();
            InterpolatorType::Pointer   interpolator3  = InterpolatorType::New();
            RegistrationType::Pointer   registration3  = RegistrationType::New();

            cout<<"Defining registration parameters..."<<endl;
            // Each component is now connected to the instance of the registration
method.
            registration1->SetMetric(metric1); registration1->SetOptimizer(optimizer1);
      registration1->SetTransform(transform1);  registration1-
>SetInterpolator(interpolator1);
            registration1->SetFixedImage(fixedROI1->GetOutput());
            registration1->SetMovingImage(movingROI1->GetOutput());
            registration1->SetFixedImageRegion(fixedROI1->GetOutput()-
>GetLargestPossibleRegion() );

            registration2->SetMetric(metric2); registration2->SetOptimizer(optimizer2);
      registration2->SetTransform(transform2);  registration2-
>SetInterpolator(interpolator2);
            registration2->SetFixedImage(fixedROI2->GetOutput());
            registration2->SetMovingImage(movingROI2->GetOutput());
            registration2->SetFixedImageRegion(fixedROI2->GetOutput()-
>GetLargestPossibleRegion() );

            registration3->SetMetric(metric3); registration3->SetOptimizer(optimizer3);
      registration3->SetTransform(transform3);  registration3-
>SetInterpolator(interpolator3);
            registration3->SetFixedImage(fixedROI3->GetOutput());
            registration3->SetMovingImage(movingROI3->GetOutput());
            registration3->SetFixedImageRegion(fixedROI3->GetOutput()-
>GetLargestPossibleRegion() );

            //  Initialize the transform
            typedef RegistrationType::ParametersType ParametersType;
            ParametersType initialParameters1( transform1->GetNumberOfParameters() );
      initialParameters1[0] = 0.0;             initialParameters1[1] = 0.0;  // Initial
offset along Y
```

```
            ParametersType initialParameters2( transform2->GetNumberOfParameters() );
      initialParameters2[0] = 0.0;             initialParameters2[1] = 0.0;  // Initial
offset along Y
            ParametersType initialParameters3( transform3->GetNumberOfParameters() );
      initialParameters3[0] = 0.0;             initialParameters3[1] = 0.0;  // Initial
offset along Y

            registration1->SetInitialTransformParameters( initialParameters1 );
            registration2->SetInitialTransformParameters( initialParameters2 );
            registration3->SetInitialTransformParameters( initialParameters3 );

            double maxStep = 1.00;
            double minStep = 0.001;
            int numIters = 500;

            optimizer1->SetMaximumStepLength( maxStep );    optimizer1-
>SetMinimumStepLength( minStep );  optimizer1->SetNumberOfIterations( numIters );
            optimizer2->SetMaximumStepLength( maxStep );    optimizer2-
>SetMinimumStepLength( minStep );  optimizer2->SetNumberOfIterations( numIters );
            optimizer3->SetMaximumStepLength( maxStep );    optimizer3-
>SetMinimumStepLength( minStep );  optimizer3->SetNumberOfIterations( numIters );

            cout<<"Registering..."<<endl;
            try
            {
                  registration1->Update();
                  registration2->Update();
                  registration3->Update();
            }
            catch( itk::ExceptionObject & err )
            {
                  std::cerr << "ExceptionObject caught !" << std::endl;
                  std::cerr << err << std::endl;
                  delete inputFilename;
                  return EXIT_FAILURE;
            }

            ParametersType finalParameters1 = registration1-
>GetLastTransformParameters();
            ParametersType finalParameters2 = registration2-
>GetLastTransformParameters();
            ParametersType finalParameters3 = registration3-
>GetLastTransformParameters();



            double averageX =
(finalParameters1[0]+finalParameters2[0]+finalParameters3[0])/3;
            double averageY =
(finalParameters1[1]+finalParameters2[1]+finalParameters3[1])/3;
            ParametersType avgFinalParameters;
            avgFinalParameters[0]=averageX;
            avgFinalParameters[1]=averageY;

            std::cout << "final parameters:" << std::endl;
            std::cout << finalParameters1 << std::endl;
            std::cout << finalParameters2 << std::endl;
            std::cout << finalParameters3 << std::endl;
```

```
		std::cout << avgFinalParameters << std::endl;

		TransformType::Pointer      finalTransform     = TransformType::New();
		finalTransform->SetParameters( avgFinalParameters );

		try
		{
			typedef itk::ResampleImageFilter<CharImageType, CharImageType >
ResampleFilterType;
			ResampleFilterType::Pointer resampler = ResampleFilterType::New();
			resampler->SetInput( caster->GetOutput() );
			resampler->SetTransform( finalTransform );
			resampler->SetSize( zeroForceReader->GetOutput()-
>GetLargestPossibleRegion().GetSize() );
			resampler->SetOutputOrigin(  zeroForceReader->GetOutput()-
>GetOrigin() );
			resampler->SetOutputSpacing( zeroForceReader->GetOutput()-
>GetSpacing() );
			resampler->SetOutputDirection( zeroForceReader->GetOutput()-
>GetDirection() );
			resampler->SetDefaultPixelValue( 0 );
			resampler->Update();

			stringstream outputBeadNames;
			outputBeadNames << inputFilename << "_redBeads_resampled" << t <<
".tif" ;

			typedef itk::ImageFileWriter<CharImageType> WriterType;
			WriterType::Pointer     writer =  WriterType::New();
			writer->SetFileName(outputBeadNames.str().c_str());
			writer->SetImageIO(tiffIO);
			writer->SetInput( resampler->GetOutput()   );
			writer->Update();
		}

		catch( itk::ExceptionObject & err )
		{
			std::cerr << "ExceptionObject caught !" << std::endl;
			std::cerr << err << std::endl;
			delete inputFilename;
			return EXIT_FAILURE;
		}
	}

	delete inputFilename;
	system("PAUSE");
	return EXIT_SUCCESS;

}



//Functions~~~
```

```
}
//Round uses the generally accepted method of rounding floats to integers in C++, using
floor and adding 0.5 to simulate rounding.
```

B.2.5 Traction force microscopy analysis code

```
//including all necessary headers:
#include "itkImage.h"
#include <iostream>
#include <stdio.h>
#include <string>
#include <math.h>
#include "itkImageRegionConstIterator.h"
#include "itkImageRegionIterator.h"
#include "itkImageRegionIteratorWithIndex.h"
#include "itkImageFileReader.h"
#include "itkImageFileWriter.h"
#include "itkTIFFImageIO.h"
#include "itkLineIterator.h"
#include "itkImageSeriesWriter.h"

#include "itkRescaleIntensityImageFilter.h"
#include "itkHistogramMatchingImageFilter.h"
#include "itkFEMRegistrationFilter.h"
#include "itkExtractImageFilter.h"
#include "itkMaskImageFilter.h"
#include "itkConnectedThresholdImageFilter.h"
#include "itkCurvatureFlowImageFilter.h"
#include "itkCastImageFilter.h"
#include <itkRGBPixel.h>
#include "itkBlobSpatialObject.h"
#include "itkSpatialObjectPoint.h"
#include "itkBinaryImageToLabelMapFilter.h"
#include "itkAddImageFilter.h"
#include "itkShiftScaleImageFilter.h"
#include "itkFFTWForwardFFTImageFilter.h"
#include "itkFFTWInverseFFTImageFilter.h"



//Namespace std will make it easier to use cout and cin functions
using namespace std;

//Defining types globally for easy access and troubleshooting

typedef itk::BlobSpatialObject<2> BlobType;

//~~~
```

```
typedef itk::ImageFileReader<ImageType3D>                    ReaderType3D;


//~~~
```

```
typedef itk::ExtractImageFilter< ImageType3D, ImageType2D >
                          ExtractFilterType;
typedef itk::CastImageFilter< FloatImageType, ImageType2D >
                          CastingFilterType;
typedef itk::CurvatureFlowImageFilter< FloatImageType, FloatImageType >
                    CurvatureFlowImageFilterType;
typedef itk::ConnectedThresholdImageFilter< FloatImageType, FloatImageType>
               ConnectedFilterType;
typedef itk::MaskImageFilter< FloatImageType, FloatImageType >
                    MaskFilterType;
typedef itk::MaskImageFilter< DispImage, FloatImageType >
                          DispMaskFilterType;

typedef itk::RGBPixel<unsigned char>
                                        RGBPixelType;
typedef itk::Image<RGBPixelType, 2 >
                                        RGBImageType;
typedef RGBImageType::IndexType
                                              RGBIndexType;
typedef itk::ImageRegionIterator< RGBImageType >
                          RGBIteratorType;
typedef itk::ImageFileWriter<RGBImageType>
                                        RGBWriterType;
typedef itk::ExtractImageFilter< ImageType3D, RGBImageType >
                    RGBExtractFilterType;


// For Binary to labeled map:
typedef itk::BinaryImageToLabelMapFilter<ImageType2D>
                    BinaryImageToLabelMapFilterType;


//function
prototypes~~~
```

```
double GreensFunction(int IndexI, int IndexJ, double K1, double K2);
double Matrix(int IndexI, int IndexJ, double K1, double K2);
double FourierTraction(int XorY, int XCoord, int YCoord, int Xmax, int Ymax,
DispImage::Pointer FFTDisplacement);


int main()
{
	//First I will make a variable for the input filename, which will be put in at
command line using the cin function
	char* inputFilename = new char[50];

	cout << "Please enter the stack's file name (must be a TIFF file)." << endl;
	cout << "WARNING:: Entering more than 50 characters will cause an error." << endl;
	cout << "Input stack file name: ";
	cin >> inputFilename;

	int flag = 1;
	cout << endl << "How many bead channels do you have?" << endl;
	cin >> flag;

	stringstream memNames;
	memNames << inputFilename << "_mem.tif" ;
	stringstream bead1Names;
	bead1Names << inputFilename << "_redBeads.tif" ;
	stringstream bead2Names;
	bead2Names << inputFilename << "_blueBeads.tif" ;
	stringstream fixedBead1Names;
	fixedBead1Names << inputFilename << "_redBeads_zeroForce.tif" ;
	stringstream fixedBead2Names;
	fixedBead2Names << inputFilename << "_blueBeads_zeroForce.tif" ;
	stringstream maskNames;
	maskNames << inputFilename << "_memMask.tif" ;

	cout<<memNames.str().c_str()<<endl;

	ReaderType3D::Pointer memReader = ReaderType3D::New();
			//creating the reader
    memReader->SetFileName( memNames.str().c_str() );
			//the reader will now look for the file name input by the user
	ReaderType3D::Pointer bead1Reader = ReaderType3D::New();
			//creating the reader
    bead1Reader->SetFileName( bead1Names.str().c_str() );
			//the reader will now look for the file name input by the user
	ReaderType3D::Pointer bead2Reader = ReaderType3D::New();
			//creating the reader
    bead2Reader->SetFileName( bead2Names.str().c_str() );
			//the reader will now look for the file name input by the user
	ReaderType2D::Pointer fixedBead1Reader = ReaderType2D::New();
		//creating the reader
    fixedBead1Reader->SetFileName( fixedBead1Names.str().c_str() );
		//the reader will now look for the file name input by the user
	ReaderType2D::Pointer fixedBead2Reader = ReaderType2D::New();
		//creating the reader
    fixedBead2Reader->SetFileName( fixedBead2Names.str().c_str() );
		//the reader will now look for the file name input by the user
```

```
   ReaderType3D::Pointer maskReader = ReaderType3D::New();
                //creating the reader
maskReader->SetFileName( maskNames.str().c_str() );
                //the reader will now look for the file name input by the user

   try
{
          memReader->Update();// reader will attempt to update with inputted filename
          bead1Reader->Update();
          fixedBead1Reader->Update();
          if (flag == 2)
          {
                 bead2Reader->Update();
                 fixedBead2Reader->Update();
          }
          maskReader->Update();
}
   catch (itk::ExceptionObject &e)
   {
          cerr << "Image not found." << endl << e << endl;
                        // if file cannot be found, program will exit
          return EXIT_FAILURE;
}

   cout << "Files found." << endl;

   RescalerType::Pointer fixedBead1Rescaler = RescalerType::New();
   RescalerType::Pointer fixedBead2Rescaler = RescalerType::New();
   const double desiredMinimum =  0.0;
   const double desiredMaximum =  255.0;

   fixedBead1Rescaler->SetInput(fixedBead1Reader->GetOutput() );
   fixedBead1Rescaler->SetOutputMinimum( desiredMinimum );
   fixedBead1Rescaler->SetOutputMaximum( desiredMaximum );
   fixedBead1Rescaler->UpdateLargestPossibleRegion();

   if(flag==2)
   {
          fixedBead2Rescaler->SetInput(fixedBead2Reader->GetOutput() );
          fixedBead2Rescaler->SetOutputMinimum( desiredMinimum );
          fixedBead2Rescaler->SetOutputMaximum( desiredMaximum );
          fixedBead2Rescaler->UpdateLargestPossibleRegion();
   }

   // Setting up the registration filter
   RegistrationType::Pointer memRegistrationFilter = RegistrationType::New();
memRegistrationFilter->SetMaxLevel(1);
memRegistrationFilter->SetUseNormalizedGradient( true );
memRegistrationFilter->ChooseMetric( 0 );

   RegistrationType::Pointer bead1RegistrationFilter = RegistrationType::New();
bead1RegistrationFilter->SetMaxLevel(1);
bead1RegistrationFilter->SetUseNormalizedGradient( true );
bead1RegistrationFilter->ChooseMetric( 0 );

   RegistrationType::Pointer bead2RegistrationFilter = RegistrationType::New();
   if (flag == 2)
   {
```

```
            bead2RegistrationFilter->SetMaxLevel(1);
            bead2RegistrationFilter->SetUseNormalizedGradient( true );
            bead2RegistrationFilter->ChooseMetric( 0 );
      }

      // Many of the following parameters were not changed from the example
registration.
      // I did change meshpixels around, but decided that 4 was the perfect balance of
speed vs. resolution
      // Number of iterations is 15 -- again to balance between speed and accuracy.
      // Elasticity and rho, while ITK recommended E/p to be roughly equal to 1, I found
that having a high elasticity and a low rho allowed for more meaningful strain maps.
      int meshPixels;
      meshPixels = 10;                                  // affects size of finite elements
(and pixel resolution of strain maps).  low values EXTREMELY slow

      unsigned int maxiters = 40;
      float        E = 10000;                  //weight parameter for elasticity matrix
      float        p = 100;                    //weight parameter for mass matrix
      float          gamma = 100;        //weight parameter for how similar the two
images should be
      memRegistrationFilter->SetElasticity(E, 0);

      memRegistrationFilter->SetRho(p, 0);

      memRegistrationFilter->SetGamma(gamma, 0);

      memRegistrationFilter->SetAlpha(1.);
      memRegistrationFilter->SetMaximumIterations( maxiters, 0 );
      memRegistrationFilter->SetMeshPixelsPerElementAtEachResolution(meshPixels, 0);

      memRegistrationFilter->SetWidthOfMetricRegion(1, 0);
      memRegistrationFilter->SetNumberOfIntegrationPoints(3, 0);
      memRegistrationFilter->SetDoLineSearchOnImageEnergy( 0 );
      memRegistrationFilter->SetTimeStep(5.);
      memRegistrationFilter->SetEmployRegridding(false);
      memRegistrationFilter->SetUseLandmarks(false);

      bead1RegistrationFilter->SetElasticity(E, 0);

      bead1RegistrationFilter->SetRho(p, 0);

      bead1RegistrationFilter->SetGamma(gamma, 0);

      bead1RegistrationFilter->SetAlpha(1.);
      bead1RegistrationFilter->SetMaximumIterations( maxiters, 0 );
      bead1RegistrationFilter->SetMeshPixelsPerElementAtEachResolution(meshPixels, 0);

      bead1RegistrationFilter->SetWidthOfMetricRegion(1, 0);
      bead1RegistrationFilter->SetNumberOfIntegrationPoints(3, 0);
      bead1RegistrationFilter->SetDoLineSearchOnImageEnergy( 0 );
      bead1RegistrationFilter->SetTimeStep(20.);
      bead1RegistrationFilter->SetEmployRegridding(false);
      bead1RegistrationFilter->SetUseLandmarks(false);

      if(flag == 2)
      {
```

```
		bead2RegistrationFilter->SetElasticity(E, 0);

		bead2RegistrationFilter->SetRho(p, 0);

		bead2RegistrationFilter->SetGamma(gamma, 0);

		bead2RegistrationFilter->SetAlpha(1.);
		bead2RegistrationFilter->SetMaximumIterations( maxiters, 0 );
		bead2RegistrationFilter-
>SetMeshPixelsPerElementAtEachResolution(meshPixels, 0);
		bead2RegistrationFilter->SetWidthOfMetricRegion(1, 0);
		bead2RegistrationFilter->SetNumberOfIntegrationPoints(2, 0);
		bead2RegistrationFilter->SetDoLineSearchOnImageEnergy( 0 );
		bead2RegistrationFilter->SetTimeStep(1.);
		bead2RegistrationFilter->SetEmployRegridding(false);
		bead2RegistrationFilter->SetUseLandmarks(true);
	}

	// Create the material properties
	itk::fem::MaterialLinearElasticity::Pointer mat;
	mat = itk::fem::MaterialLinearElasticity::New();
	mat->SetGlobalNumber(0);
	mat->SetYoungsModulus(bead1RegistrationFilter->GetElasticity()); // Young's
modulus of the membrane
	mat->SetCrossSectionalArea(1.0);                              // Cross-sectional
area
	mat->SetThickness(1.0);                                       // Thickness
	mat->SetMomentOfInertia(1.0);                                 // Moment of inertia
	mat->SetPoissonsRatio(0.5);                                   // Poisson's ratio --
tissues are mostly composed of water, which is incompressible.  So 0.5 makes sense here.
	mat->SetDensityHeatProduct(1.0);                              // Density-Heat
capacity product

	// Create the element type
	ElementType::Pointer e1=ElementType::New();
	e1->SetMaterial(mat.GetPointer());
	memRegistrationFilter->SetElement(e1.GetPointer());
	memRegistrationFilter->SetMaterial(mat);
	bead1RegistrationFilter->SetElement(e1.GetPointer());
	bead1RegistrationFilter->SetMaterial(mat);
	if(flag==2)
	{
		bead2RegistrationFilter->SetElement(e1.GetPointer());
		bead2RegistrationFilter->SetMaterial(mat);
	}

	//Getting dimensions of input image
	ImageType3D::RegionType inputRegion = memReader->GetOutput()-
>GetLargestPossibleRegion();
	ImageType3D::SizeType inputSize = inputRegion.GetSize();

	//the size of the slices will be identical to the input images, except with a size
of 0 in the z-direction
	ImageType3D::SizeType desiredSize = inputRegion.GetSize();
	desiredSize[2] = 0;
```

```
	//this allows the user to choose how many slices to the program on, so you need
not run the entire registration (which takes a long time) if you
	//are debugging or adjusting parameters.  There is also a checkpoint to assure the
range is within the actual size of the image.
	int range = inputSize[2];

	// Finally the for loop, which will for iterative registrations, is setup.  The
range input by the user is used as the stopping point here.
	for(unsigned int j = 0; j < range; j++)
	{
		//Cout number j so the user will know how far along their program is
		cout << "Starting membrane registration #" << j << endl;

		// ExtractFilterType will allow for the extraction of 2D slices from the 3D
stack read in by the reader.  Two extractors are necessary
		// because the registration filter requires two images for a registration.
		// Initializing them now makes sense because these will be constant
throughout the registration loop.
		ExtractFilterType::Pointer memExtractor1 = ExtractFilterType::New();
		memExtractor1->InPlaceOn();
		memExtractor1->SetDirectionCollapseToSubmatrix();
		ExtractFilterType::Pointer memExtractor2 = ExtractFilterType::New();
		memExtractor2->InPlaceOn();
		memExtractor2->SetDirectionCollapseToSubmatrix();
		RGBExtractFilterType::Pointer RGBextractor = RGBExtractFilterType::New();
		RGBextractor->InPlaceOn();
		RGBextractor->SetDirectionCollapseToSubmatrix();


		ExtractFilterType::Pointer bead1Extractor = ExtractFilterType::New();
		bead1Extractor->InPlaceOn();
		bead1Extractor->SetDirectionCollapseToSubmatrix();

		ExtractFilterType::Pointer bead2Extractor = ExtractFilterType::New();

		if(flag==2)
		{
			bead2Extractor->InPlaceOn();
			bead2Extractor->SetDirectionCollapseToSubmatrix();
		}

		ExtractFilterType::Pointer maskExtractor = ExtractFilterType::New();
		maskExtractor->InPlaceOn();
		maskExtractor->SetDirectionCollapseToSubmatrix();

		//Extractor1 will get slice j while extractor2 gets slice (j+1).  These can
then be plugged into the registration filter.
		ImageType3D::IndexType extractorStart1 = inputRegion.GetIndex();
		extractorStart1[2] = j;
		ImageType3D::RegionType desiredRegion1;
		desiredRegion1.SetSize( desiredSize );
		desiredRegion1.SetIndex( extractorStart1 );
		memExtractor1->SetExtractionRegion( desiredRegion1 );
		memExtractor1->SetInput( memReader->GetOutput() );
		memExtractor1->Update();
		RGBextractor->SetExtractionRegion( desiredRegion1 );
		RGBextractor->SetInput( memReader->GetOutput() );
		RGBextractor->Update();
```

```
bead1Extractor->SetExtractionRegion( desiredRegion1 );
bead1Extractor->SetInput( bead1Reader->GetOutput() );
bead1Extractor->Update();
if(flag == 2)
{
	bead2Extractor->SetExtractionRegion( desiredRegion1 );
	bead2Extractor->SetInput( bead2Reader->GetOutput() );
	bead2Extractor->Update();
}

maskExtractor->SetExtractionRegion( desiredRegion1 );
maskExtractor->SetInput( maskReader->GetOutput() );
maskExtractor->Update();

ImageType3D::IndexType desiredStart2 = inputRegion.GetIndex();
desiredStart2[2] = j + 1;
ImageType3D::RegionType desiredRegion2;
desiredRegion2.SetSize( desiredSize );
desiredRegion2.SetIndex( desiredStart2 );
memExtractor2->SetExtractionRegion( desiredRegion2 );
memExtractor2->SetInput( memReader->GetOutput() );
memExtractor2->Update();

//WriteCharImage(memExtractor1->GetOutput(), "test1.tif");
//WriteCharImage(memExtractor2->GetOutput(), "test2.tif");
//WriteCharImage(beadExtractor1->GetOutput(), "test3.tif");
//WriteCharImage(beadExtractor2->GetOutput(), "test4.tif");
//WriteRGBImage(RGBextractor->GetOutput(), "testRGB.tif");

RescalerType::Pointer memRescaler1 = RescalerType::New();
RescalerType::Pointer memRescaler2 = RescalerType::New();
RescalerType::Pointer bead1Rescaler = RescalerType::New();
RescalerType::Pointer bead2Rescaler = RescalerType::New();
RescalerType::Pointer maskRescaler = RescalerType::New();

memRescaler1->SetInput(memExtractor1->GetOutput() );
memRescaler2->SetInput(memExtractor2->GetOutput() );
memRescaler1->SetOutputMinimum( desiredMinimum );
memRescaler1->SetOutputMaximum( desiredMaximum );
memRescaler1->UpdateLargestPossibleRegion();
memRescaler2->SetOutputMinimum( desiredMinimum );
memRescaler2->SetOutputMaximum( desiredMaximum );
memRescaler2->UpdateLargestPossibleRegion();

bead1Rescaler->SetInput(bead1Extractor->GetOutput() );
bead1Rescaler->SetOutputMinimum( desiredMinimum );
bead1Rescaler->SetOutputMaximum( desiredMaximum );
bead1Rescaler->UpdateLargestPossibleRegion();

if(flag==2)
{
	bead2Rescaler->SetInput(bead2Extractor->GetOutput() );
	bead2Rescaler->SetOutputMinimum( desiredMinimum );
	bead2Rescaler->SetOutputMaximum( desiredMaximum );
	bead2Rescaler->UpdateLargestPossibleRegion();
}
```

```
		maskRescaler->SetInput(maskExtractor->GetOutput() );
		maskRescaler->SetOutputMinimum( desiredMinimum );
		maskRescaler->SetOutputMaximum( desiredMaximum );
		maskRescaler->UpdateLargestPossibleRegion();

		// The histogram filter helps match the images -- it was taken from the
example registration provided in the ITK documentation
		// Parameters for the historgram filter have not been changed from the
example -- these seemed to work well.
		HEFilterType::Pointer memIntensityEqualizeFilter = HEFilterType::New();
		memIntensityEqualizeFilter->SetReferenceImage( memRescaler1->GetOutput() );
		memIntensityEqualizeFilter->SetInput( memRescaler2->GetOutput() );
		memIntensityEqualizeFilter->SetNumberOfHistogramLevels(100);
		memIntensityEqualizeFilter->SetNumberOfMatchPoints(15);
		memIntensityEqualizeFilter->ThresholdAtMeanIntensityOn();
		memIntensityEqualizeFilter->Update();

		// Setting the images for registration filter and running the registration
		memRegistrationFilter->SetFixedImage( memRescaler1->GetOutput() );
		memRegistrationFilter->SetMovingImage( memIntensityEqualizeFilter-
>GetOutput() );
		memRegistrationFilter->Modified();
		memRegistrationFilter->RunRegistration();

		cout << "Starting bead1 registration #" << j << endl;

		HEFilterType::Pointer bead1IntensityEqualizeFilter = HEFilterType::New();
		HEFilterType::Pointer bead2IntensityEqualizeFilter = HEFilterType::New();
		bead1IntensityEqualizeFilter->SetReferenceImage( fixedBead1Rescaler-
>GetOutput() );
		bead1IntensityEqualizeFilter->SetInput( bead1Rescaler->GetOutput() );
		bead1IntensityEqualizeFilter->SetNumberOfHistogramLevels(100);
		bead1IntensityEqualizeFilter->SetNumberOfMatchPoints(15);
		bead1IntensityEqualizeFilter->ThresholdAtMeanIntensityOn();
		bead1IntensityEqualizeFilter->Update();

		// Setting the images for registration filter and running the registration
		bead1RegistrationFilter->SetFixedImage( fixedBead1Rescaler->GetOutput() );
		bead1RegistrationFilter->SetMovingImage( bead1IntensityEqualizeFilter-
>GetOutput() );
		bead1RegistrationFilter->Modified();
		bead1RegistrationFilter->RunRegistration();

		if(flag == 2)
		{
			cout << "Starting bead2 registration #" << j << endl;
			bead2IntensityEqualizeFilter->SetReferenceImage( fixedBead2Rescaler-
>GetOutput() );
			bead2IntensityEqualizeFilter->SetInput( bead2Rescaler->GetOutput()
);
			bead2IntensityEqualizeFilter->SetNumberOfHistogramLevels(100);
			bead2IntensityEqualizeFilter->SetNumberOfMatchPoints(15);
			bead2IntensityEqualizeFilter->ThresholdAtMeanIntensityOn();
			bead2IntensityEqualizeFilter->Update();

			// Setting the images for registration filter and running the
registration
```

```
                    bead2RegistrationFilter->SetFixedImage( fixedBead2Rescaler-
>GetOutput() );
                    bead2RegistrationFilter->SetMovingImage(
bead2IntensityEqualizeFilter->GetOutput() );
                    bead2RegistrationFilter->Modified();
                    bead2RegistrationFilter->RunRegistration();
             }

             // The X- and Y- displacements from the registration need to be written and
stored.  deltaX and deltaY are initialized to do so.  the initial displacement image is
also initialized
             FloatImageType::Pointer memDeltaX = FloatImageType::New();
             FloatImageType::Pointer memDeltaY = FloatImageType::New();
             memDeltaX->SetRegions(memExtractor1->GetOutput()-
>GetLargestPossibleRegion());
             memDeltaX->Allocate();
             memDeltaY->SetRegions(memExtractor1->GetOutput()-
>GetLargestPossibleRegion());
             memDeltaY->Allocate();

             FloatImageType::Pointer beadDeltaX = FloatImageType::New();
             FloatImageType::Pointer beadDeltaY = FloatImageType::New();
             beadDeltaX->SetRegions(fixedBead1Reader->GetOutput()-
>GetLargestPossibleRegion());
             beadDeltaX->Allocate();
             beadDeltaY->SetRegions(fixedBead1Reader->GetOutput()-
>GetLargestPossibleRegion());
             beadDeltaY->Allocate();

             DispImage::Pointer beadDisplacement = DispImage::New();
             beadDisplacement->SetRegions(bead1RegistrationFilter-
>GetDisplacementField()->GetLargestPossibleRegion());
             beadDisplacement->Allocate();
             //AddDisplacementsFilter::Pointer addDisps = AddDisplacementsFilter::New();
             //ScaleDisplacementsFilter::Pointer scaleDisps =
ScaleDisplacementsFilter::New();
             if(flag == 2)
             {
                    DispIteratorType  beadItD1(  bead1RegistrationFilter-
>GetDisplacementField(),  bead1RegistrationFilter->GetDisplacementField()-
>GetLargestPossibleRegion());
                    DispIteratorType  beadItD2(  bead2RegistrationFilter-
>GetDisplacementField(),  bead2RegistrationFilter->GetDisplacementField()-
>GetLargestPossibleRegion());
                    DispIteratorType  beadItDout(  beadDisplacement,
bead1RegistrationFilter->GetDisplacementField()->GetLargestPossibleRegion());
                    for(beadItD1.GoToBegin(), beadItD2.GoToBegin(),
beadItDout.GoToBegin(); !beadItD1.IsAtEnd(); ++beadItD1, ++beadItD2, ++beadItDout )
                    {
                           float x1 = (beadItD1.Get())[0];
                           float x2 = (beadItD2.Get())[0];
                           float y1 = (beadItD1.Get())[1];
                           float y2 = (beadItD2.Get())[1];
                           float xavg = (x1+x2)/2;
                           float yavg = (y1+y2)/2;
                           itk::Vector<float,2> avgDisp;
                           avgDisp[0] = xavg;
                           avgDisp[1] = yavg;
```

```
                        beadItDout.Set(avgDisp);
                }
                //addDisps->SetInput1(bead1RegistrationFilter-
>GetDisplacementField());
                //addDisps->SetInput2(bead2RegistrationFilter-
>GetDisplacementField());
                //addDisps->Update();
                //scaleDisps->SetInput(addDisps->GetOutput());
                //scaleDisps->SetScale(0.5);
                //scaleDisps->Update();
                //beadDisplacement = scaleDisps->GetOutput();
        }
        else
        {
                beadDisplacement = bead1RegistrationFilter->GetDisplacementField();
        }

        // the displacement is output as a vector image, so I will use iterators to
isolate the X- and Y-displacements.
        // as the iterator moves through the displacement images, it sets the value
of deltaX to the x component of the vector and deltaY to the y
        // component of the vector.
        FloatIteratorType memItX(  memDeltaX, memExtractor1->GetOutput()-
>GetLargestPossibleRegion() );
        FloatIteratorType memItY(  memDeltaY, memExtractor1->GetOutput()-
>GetLargestPossibleRegion() );
        DispIteratorType memItD ( memRegistrationFilter->GetDisplacementField(),
memRegistrationFilter->GetDisplacementField()->GetLargestPossibleRegion());
        for(memItX.GoToBegin(), memItY.GoToBegin(), memItD.GoToBegin();
!memItD.IsAtEnd(); ++memItX, ++memItY, ++memItD )
        {
                memItX.Set( (memItD.Get())[0] );

                memItY.Set( (memItD.Get())[1] );
        }

        FloatIteratorType beadItX(  beadDeltaX, fixedBead1Reader->GetOutput()-
>GetLargestPossibleRegion() );
        FloatIteratorType beadItY(  beadDeltaY, fixedBead1Reader->GetOutput()-
>GetLargestPossibleRegion() );
        DispIteratorType beadItD ( beadDisplacement, bead1RegistrationFilter-
>GetDisplacementField()->GetLargestPossibleRegion());
        for(beadItX.GoToBegin(), beadItY.GoToBegin(), beadItD.GoToBegin();
!beadItD.IsAtEnd(); ++beadItX, ++beadItY, ++beadItD )
        {
                beadItX.Set( (beadItD.Get())[0] );

                beadItY.Set( (beadItD.Get())[1] );
        }

        MaskFilterType::Pointer memXMaskFilter = MaskFilterType::New();
        MaskFilterType::Pointer memYMaskFilter = MaskFilterType::New();
        //MaskFilterType::Pointer beadXMaskFilter = MaskFilterType::New();
        //MaskFilterType::Pointer beadYMaskFilter = MaskFilterType::New();
        memXMaskFilter->SetInput(memDeltaX);
        memXMaskFilter->SetMaskImage( maskRescaler->GetOutput() );
        memXMaskFilter->Update();
```

```
		memYMaskFilter->SetInput(memDeltaY);
		memYMaskFilter->SetMaskImage( maskRescaler->GetOutput() );
		memYMaskFilter->Update();

		//beadXMaskFilter->SetInput(beadDeltaX);
		//beadXMaskFilter->SetMaskImage( maskRescaler->GetOutput() );
		//beadXMaskFilter->Update();

		//beadYMaskFilter->SetInput(beadDeltaY);
		//beadYMaskFilter->SetMaskImage( maskRescaler->GetOutput() );
		//beadYMaskFilter->Update();

		// stringstreams are set up for sequential output file names, using j as
the number to be printed.
		// the WriteImage function was created to make this process easier, as I am
writing multiple images.  See below main function for explanation.
		stringstream memDeltaXnames;
		memDeltaXnames << inputFilename << "_mem_deltaX0" << j << ".tif";
		WriteImage( memXMaskFilter->GetOutput(), memDeltaXnames.str().c_str() );
		stringstream memDeltaYnames;
		memDeltaYnames << inputFilename << "_mem_deltaY0" << j << ".tif";
		WriteImage( memYMaskFilter->GetOutput(), memDeltaYnames.str().c_str() );
		stringstream beadDeltaXnames;
		beadDeltaXnames << inputFilename << "_beads_deltaX0" << j << ".tif";
		WriteImage( beadDeltaX, beadDeltaXnames.str().c_str() );
		stringstream beadDeltaYnames;
		beadDeltaYnames << inputFilename << "_beads_deltaY0" << j << ".tif";
		WriteImage( beadDeltaY, beadDeltaYnames.str().c_str() );

		FloatImageType::Pointer angleCorrelationImage = FloatImageType::New();
		angleCorrelationImage->SetRegions(fixedBead1Reader->GetOutput()-
>GetLargestPossibleRegion());
		angleCorrelationImage->Allocate();

		FloatIteratorType angleCorrIt(  angleCorrelationImage, memExtractor1-
>GetOutput()->GetLargestPossibleRegion() );
		for(memItX.GoToBegin(), memItY.GoToBegin(),beadItX.GoToBegin(),
beadItY.GoToBegin(), angleCorrIt.GoToBegin(); !memItX.IsAtEnd(); ++beadItX, ++beadItY,
++memItX, ++memItY, ++angleCorrIt)
		{

			float v1x = memItX.Get();
			float v1y = memItY.Get();
			float v2x = beadItX.Get();
			float v2y = beadItY.Get();

			float norm1 = sqrt( (v1x*v1x)+(v1y*v1y) );
			float norm2 = sqrt( (v2x*v2x)+(v2y*v2y) );

			//if(norm1 > 0.2 && norm2 > 0.2)
			//{
				float correlationScore = (v1x*v2x + v1y*v2y)/( norm1*norm2 );
				angleCorrIt.Set(correlationScore);
			//}
			//else
			//{
				//angleCorrIt.Set(0);
			//}
```

```
		}

		MaskFilterType::Pointer angleMaskFilter = MaskFilterType::New();
		angleMaskFilter->SetInput(angleCorrelationImage);
		angleMaskFilter->SetMaskImage( maskRescaler->GetOutput() );
		angleMaskFilter->Update();

		stringstream angleCorrNames;
		angleCorrNames << inputFilename << "_angleCorrelation0" << j << ".tif";
		WriteImage( angleMaskFilter->GetOutput(), angleCorrNames.str().c_str() );


		RGBImageType::Pointer vectorImage = RGBImageType::New();
		vectorImage->Allocate();
		vectorImage = RGBextractor->GetOutput();
		stringstream vectorNames;
		vectorNames << inputFilename << "_vector0" << j << ".tif";

		MaskFilterType::Pointer deltaXMaskFilter = MaskFilterType::New();
		deltaXMaskFilter->SetInput(memDeltaX);
		deltaXMaskFilter->SetMaskImage( maskRescaler->GetOutput() );
		deltaXMaskFilter->Update();
		MaskFilterType::Pointer deltaYMaskFilter = MaskFilterType::New();
		deltaYMaskFilter->SetInput(memDeltaY);
		deltaYMaskFilter->SetMaskImage( maskRescaler->GetOutput() );
		deltaYMaskFilter->Update();


		RGBIteratorType vectorIt(  vectorImage, vectorImage-
>GetLargestPossibleRegion() );
		for(vectorIt.GoToBegin(); !vectorIt.IsAtEnd(); ++vectorIt)
		{
			FloatImageType::IndexType temp = vectorIt.GetIndex();
			float XValue = ((deltaXMaskFilter->GetOutput())->GetPixel(temp));
			float YValue = ((deltaYMaskFilter->GetOutput())->GetPixel(temp));

			RGBPixelType white;
			white[0] = 255;
			white[1] = 255;
			white[2] = 255;

			if ( ( (temp[0] % 14) == 0) && (((temp[1] % 14) == 0)) && ( XValue
!= 0)&&(YValue != 0) )
			{
				DrawDisplacement(vectorImage, deltaXMaskFilter->GetOutput(),
deltaYMaskFilter->GetOutput(), temp, 20, white);
			}
		}
		WriteRGBImage( vectorImage, vectorNames.str().c_str() );
		/*
		typedef itk::FFTWForwardFFTImageFilter<DispImage,DispImage> FFTtype;
		FFTtype::Pointer fft = FFTtype::New();
		fft->SetInput(beadDisplacement);
		fft->Update();

		DispImage::Pointer fftDisp = fft->GetOutput();
		DispImage::Pointer fftTraction = DispImage::New();
```

```
		fftTraction->SetRegions(fftDisp->GetLargestPossibleRegion());
		fftTraction->Allocate();

		DispIteratorType fftIt( fftDisp, fftDisp->GetLargestPossibleRegion() );
		DispIteratorType fftTractionIt( fftTraction, fftTraction-
>GetLargestPossibleRegion() );

		for(fftIt.GoToBegin(),fftTractionIt.GoToBegin(); !fftIt.IsAtEnd(); ++fftIt,
++fftTractionIt)
		{
			DispImage::PixelType tempFFTdisp = fftIt.Get();
			DispIndex tempLocation = fftIt.GetIndex();
			DispImage::PixelType tempFFTtraction;
			tempFFTtraction[0] = FourierTraction(0,
tempLocation[0],tempLocation[1],inputSize[0],inputSize[1],fftDisp);
			tempFFTtraction[1] = FourierTraction(1,
tempLocation[0],tempLocation[1],inputSize[0],inputSize[1],fftDisp);

			fftTractionIt.Set(tempFFTtraction);
		}

		typedef itk::FFTWInverseFFTImageFilter<DispImage,DispImage> IFFTtype;
		IFFTtype::Pointer ifft = IFFTtype::New();
		ifft->SetInput(fftTraction);
		ifft->Update();

		FloatImageType::Pointer tractionX = FloatImageType::New();
		FloatImageType::Pointer tractionY = FloatImageType::New();
		tractionX->SetRegions(fixedBead1Reader->GetOutput()-
>GetLargestPossibleRegion());
		tractionX->Allocate();
		tractionY->SetRegions(fixedBead1Reader->GetOutput()-
>GetLargestPossibleRegion());
		tractionY->Allocate();

		FloatIteratorType txIt( tractionX, tractionX->GetLargestPossibleRegion() );
		FloatIteratorType tyIt( tractionY, tractionY->GetLargestPossibleRegion() );
		DispIteratorType tIt( ifft->GetOutput(), ifft->GetOutput()-
>GetLargestPossibleRegion() );

		for(txIt.GoToBegin(),tyIt.GoToBegin(),tIt.GoToBegin(); !txIt.IsAtEnd();
++txIt, ++tyIt, ++tIt)
		{
			txIt.Set( tIt.Get()[0] );
			tyIt.Set( tIt.Get()[1] );
		}

		stringstream tractionXnames;
		tractionXnames << inputFilename << "_tractionX0" << j << ".tif";
		WriteImage( tractionX, tractionXnames.str().c_str() );
		stringstream tractionYnames;
		tractionYnames << inputFilename << "_tractionY0" << j << ".tif";
		WriteImage( tractionY, tractionYnames.str().c_str() );
		*/
		}


	//deallocating memory
```

```
	delete [] inputFilename;

	system("PAUSE");
	return EXIT_SUCCESS;

}



//Functions~~~
```

```
	{
		writer->Update();

    }
	catch (itk::ExceptionObject &e)
    {
		cerr << FileName << " could not be written." << endl << e << endl;   // if
there is a writing error, program will exit
		return EXIT_FAILURE;
	}
	return EXIT_SUCCESS;
}

int WriteCharImage( ImageType2D::Pointer ImageToWrite, const char* FileName ){


	WriterType2D::Pointer writer = WriterType2D::New();
	writer->SetFileName( FileName );
	writer->SetInput( ImageToWrite );

	TIFFIOType::Pointer tiffIO = TIFFIOType::New();
	//output image will be a .tif file
	writer->SetImageIO(tiffIO);

	try
	{
		writer->Update();

    }
	catch (itk::ExceptionObject &e)
    {
		cerr << FileName << " could not be written." << endl << e << endl;   // if
there is a writing error, program will exit
		return EXIT_FAILURE;
	}
	return EXIT_SUCCESS;
}

int DrawDisplacement ( RGBImageType::Pointer ImageToDrawOn, FloatImageType::Pointer
DeltaX, FloatImageType::Pointer DeltaY, FloatImageType::IndexType StartingPoint, float
Scalar, RGBPixelType Color)
{
	float X;
	float Y;
	FloatIndex temp = StartingPoint;
	ImageType2D::SizeType size = ImageToDrawOn->GetLargestPossibleRegion().GetSize();
	X = (DeltaX->GetPixel(temp))*Scalar;
	Y = (DeltaY->GetPixel(temp))*Scalar;
	FloatIndex terminalPoint;
	terminalPoint[0] = Round(temp[0] + X);
	terminalPoint[1] = Round(temp[1] + Y);
	if((temp !=
terminalPoint)&&(size[0]>terminalPoint[0])&&(size[1]>terminalPoint[1]))
	{
	FloatIndex arrowPoint1;
	FloatIndex arrowPoint2;
	float theta = 0.26179;  // 15 degrees
	float x1 = terminalPoint[0];
```

```
	float y1 = terminalPoint[1];
	float x2 = StartingPoint[0];
	float y2 = StartingPoint[1];
	float AB = sqrt(((x1-x2)*(x1-x2))+((y1-y2)*(y1-y2)));
	float v1 = (cos(theta)*(x2-x1) + sin(theta)*(y2-y1))/(AB);
	float v2 = (cos(theta)*(y2-y1) - sin(theta)*(x2-x1))/(AB);
	float v3 = (cos(-theta)*(x2-x1) + sin(-theta)*(y2-y1))/(AB);
	float v4 = (cos(-theta)*(y2-y1) - sin(-theta)*(x2-x1))/(AB);
	arrowPoint1[0] = Round(x1+5*v1);
	arrowPoint1[1] = Round(y1+5*v2);
	arrowPoint2[0] = Round(x1+5*v3);
	arrowPoint2[1] = Round(y1+5*v4);

	itk::LineIterator<RGBImageType> drawline1(ImageToDrawOn, StartingPoint,
terminalPoint); drawline1.GoToBegin();
	while (!drawline1.IsAtEnd())  {drawline1.Set(Color); ++drawline1;}

	if((size[0]>arrowPoint1[0])&&(size[1]>arrowPoint1[1]))
	{
		itk::LineIterator<RGBImageType> drawline2(ImageToDrawOn, terminalPoint,
arrowPoint1); drawline2.GoToBegin();
		while (!drawline2.IsAtEnd())  {drawline2.Set(Color); ++drawline2;}
	}

	if((size[0]>arrowPoint2[0])&&(size[1]>arrowPoint2[1]))
	{
	itk::LineIterator<RGBImageType> drawline3(ImageToDrawOn, terminalPoint,
arrowPoint2); drawline3.GoToBegin();
	while (!drawline3.IsAtEnd())  {drawline3.Set(Color); ++drawline3;}
	}

	}

	return EXIT_SUCCESS;
}

int DrawAvgDisplacement ( RGBImageType::Pointer ImageToDrawOn, double AvgDeltaX, double
AvgDeltaY, FloatImageType::IndexType StartingPoint, float Scalar, RGBPixelType Color)
{
	float X;
	float Y;
	FloatIndex temp = StartingPoint;
	ImageType2D::SizeType size = ImageToDrawOn->GetLargestPossibleRegion().GetSize();
	X = AvgDeltaX*Scalar;
	Y = AvgDeltaY*Scalar;
	FloatIndex terminalPoint;
	terminalPoint[0] = Round(temp[0] + X);
	terminalPoint[1] = Round(temp[1] + Y);
	if((temp !=
terminalPoint)&&(size[0]>terminalPoint[0])&&(size[1]>terminalPoint[1]))
	{
	FloatIndex arrowPoint1;
	FloatIndex arrowPoint2;
	float theta = 0.26179;  // 15 degrees
	float x1 = terminalPoint[0];
	float y1 = terminalPoint[1];
	float x2 = StartingPoint[0];
	float y2 = StartingPoint[1];
```

```
	float AB = sqrt(((x1-x2)*(x1-x2))+((y1-y2)*(y1-y2)));
	float v1 = (cos(theta)*(x2-x1) + sin(theta)*(y2-y1))/(AB);
	float v2 = (cos(theta)*(y2-y1) - sin(theta)*(x2-x1))/(AB);
	float v3 = (cos(-theta)*(x2-x1) + sin(-theta)*(y2-y1))/(AB);
	float v4 = (cos(-theta)*(y2-y1) - sin(-theta)*(x2-x1))/(AB);
	arrowPoint1[0] = Round(x1+5*v1);
	arrowPoint1[1] = Round(y1+5*v2);
	arrowPoint2[0] = Round(x1+5*v3);
	arrowPoint2[1] = Round(y1+5*v4);

	itk::LineIterator<RGBImageType> drawline1(ImageToDrawOn, StartingPoint,
terminalPoint); drawline1.GoToBegin();
	while (!drawline1.IsAtEnd())  {drawline1.Set(Color); ++drawline1;}

	if((size[0]>arrowPoint1[0])&&(size[1]>arrowPoint1[1]))
	{
		itk::LineIterator<RGBImageType> drawline2(ImageToDrawOn, terminalPoint,
arrowPoint1); drawline2.GoToBegin();
		while (!drawline2.IsAtEnd())  {drawline2.Set(Color); ++drawline2;}
	}

	if((size[0]>arrowPoint2[0])&&(size[1]>arrowPoint2[1]))
	{
	itk::LineIterator<RGBImageType> drawline3(ImageToDrawOn, terminalPoint,
arrowPoint2); drawline3.GoToBegin();
	while (!drawline3.IsAtEnd())  {drawline3.Set(Color); ++drawline3;}
	}

	}

	return EXIT_SUCCESS;
}


unsigned int GetCenterOfMass(unsigned int RegionLabel, unsigned int XorY,
BinaryImageToLabelMapFilterType::Pointer filter)
{
	BinaryImageToLabelMapFilterType::OutputImageType::LabelObjectType* labelObject =
filter->GetOutput()->GetNthLabelObject(RegionLabel);

	std::vector <unsigned int> coords;

	for(unsigned int pixelId = 0; pixelId < labelObject->Size(); pixelId++)
		{
	    coords.push_back(labelObject->GetIndex(pixelId)[XorY]);
		}

	unsigned int sumCoords = 0;
	for(std::vector<unsigned int>::iterator j = coords.begin(); j!=coords.end(); ++j)
	{sumCoords += *j;}

	unsigned int size = coords.size();
	unsigned int centerOfMass = sumCoords / size;

	return centerOfMass;
}
```

```
double GetAverage(unsigned int RegionLabel, FloatImageType::Pointer DisplacementImage,
BinaryImageToLabelMapFilterType::Pointer filter)
{
	BinaryImageToLabelMapFilterType::OutputImageType::LabelObjectType* labelObject =
filter->GetOutput()->GetNthLabelObject(RegionLabel);

	std::vector <FloatIndex> coords;

	for(unsigned int pixelId = 0; pixelId < labelObject->Size(); pixelId++)
	{
		FloatIndex temp;
		temp[0] = labelObject->GetIndex(pixelId)[0];
		temp[1] = labelObject->GetIndex(pixelId)[1];
		coords.push_back(temp);
	}

	unsigned int size = coords.size();

	double sumDisplacement = 0;

	for(unsigned int p = 0; p < size; p++)
	{
		sumDisplacement += DisplacementImage->GetPixel(coords.at(p));
	}

	double avgDisplacement = sumDisplacement / size;

	return avgDisplacement;
}
```

B.3 TISSUE DEFORMATION TRACKING

This section outlines the microsurgery for a “belly-up” preparation, to image the ventral morphogenesis of an embryo, as well as the image analysis for tissue deformation tracking.

B.3.1 “Belly-up” microsurgical preparation.

This is a fairly straightforward microsurgical preparation (Fig. 49). The embryos are devitellinized at Stage 13 and wounded on their dorsal side in 1/3X MBS supplemented with antibiotic. It is important to REMOVE cells while creating a wound, and not just cut. Immediately after wounding, mount the dorsal side of the embryo onto fibronectin and gently compress with a glass coverslip. Let the embryo sit for 2 hours. Then remove the glass coverslip and begin filming.

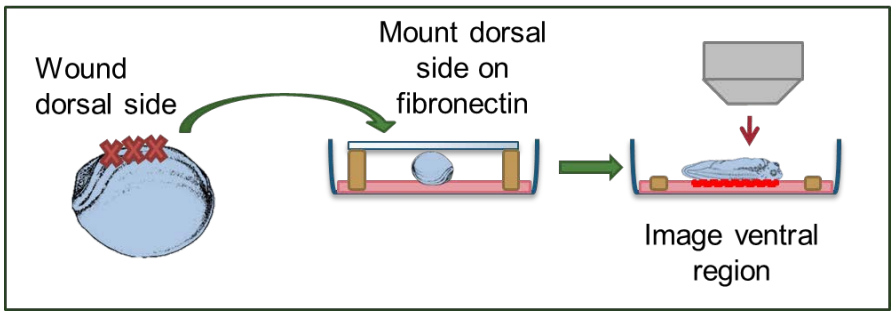

Figure 49. Schematic of “belly-up” microsurgical preparation

B.3.2 Tissue deformation tracking image analysis

Time lapses of ventral elongation were input into the following code. A grid of points is generated by the user, using inputted values for the center of the grid, the length and width of the grid and the number of grid squared. Then each consecutive pair of frames is registered, using an FEM-based registration filter, which calculates a displacement map to map one frame onto the next. The calculated displacements are then applied to the grid points and then overlaid onto the image.

One important note: because the drawn grid points need to be discrete values (i.e., whole numbers) but the calculated displacements are often fractions of pixels, in order to maintain accuracy in the deformation tracking, I needed to store the true displaced grid points (i.e., decimal values) and rounded grid points for drawing purposes in separate arrays. This made a huge difference in the precision of the deformation tracking.

The code, which is written in C++ implementing open source filters of ITK, is shown below. The input image and number of frames to analyze is selected by the user, as is the number of pixels in a mesh element (typically values from 4-8 give the best results). The final output is a text file that is properly formatted open in Microsoft Excel.

B.3.3 Tissue deformation tracking code

```
//including all necessary headers:
#include "itkImage.h"
#include <iostream>
#include <stdio.h>
#include <string>
#include <math.h>
```

```
#include "itkImageRegionConstIterator.h"
#include "itkImageRegionIterator.h"
#include "itkImageRegionIteratorWithIndex.h"
#include "itkImageFileReader.h"
#include "itkImageFileWriter.h"
#include "itkTIFFImageIO.h"
#include "itkLineIterator.h"
#include "itkImageSeriesWriter.h"

#include "itkRescaleIntensityImageFilter.h"
#include "itkHistogramMatchingImageFilter.h"
#include "itkFEMRegistrationFilter.h"
#include "itkExtractImageFilter.h"
#include "itkMaskImageFilter.h"
#include "itkConnectedThresholdImageFilter.h"
#include "itkCurvatureFlowImageFilter.h"
#include "itkCastImageFilter.h"
#include <itkRGBPixel.h>
#include "itkBlobSpatialObject.h"
#include "itkSpatialObjectPoint.h"



//Namespace std will make it easier to use cout and cin functions
using namespace std;

//Defining types globally for easy access and troubleshooting

typedef itk::BlobSpatialObject<2> BlobType;

//~~~
```

```
//~~~
```

```
int Round(float Number);
// Round is created because the displacements are float type, but the indices need to be
integers.

int WriteImage( FloatImageType::Pointer ImageToWrite, const char* FileName  );
// WriteImage makes it easier to print images to file

int DrawBox( RGBImageType::Pointer ImageToDrawOn, int x1, int y1, int x2, int y2, int x3,
int y3, int x4, int y4 );
int DrawGrid( RGBImageType::Pointer ImageToDrawOn, vector<int> ArrayOfXIndices,
vector<int> ArrayOfYIndices, vector<int> ListOfQuads  );
// DrawBox takes four points and creates one quadrilateral of the 3x3 distortion diagram
// DrawGrid uses DrawBox 9 times to make the entire grid

void SetupIndices (int PointNumber, DispIndex* ArrayOfIndices, float*
ArrayOfIndicesActual, DispIndex Index, int XCoord, int YCoord );
//SetupIndices simply takes integers and plugs them the pre-existing array of ITK
indices.  I found it easier to create a simple function for this
//than to repeat the code so many times.  Having an array of indices makes it much easier
to generate the grid, because I can pass the array in as a
//single variable.

float UpdateIndices (vector<float> ArrayOfXIndices, vector<float> ArrayOfYIndices,
DispImage::Pointer DisplacementImage, int PointNumber, int XorY);
//UpdateIndices takes the displacements calculated from the registration filter and adds
them to the pre-existing array of indices to generate the distorted grid.

void PrintData(vector<int> ArrayOfXIndices, vector<int> ArrayOfYIndices, int
RegistrationNumber, int WhichSquare, int A, int B, int C, int D, FILE* textFilePointer,
vector<float> MovingAreas);

void PrintAllData( vector<int> ArrayOfXIndices, vector<int> ArrayOfYIndices, vector<int>
ListOfQuads, int RegistrationNumber, FILE* textFilePointer, vector<float> MovingAreas);
//PrintData calculates the area and length-to-width ratio of a single quadrilateral given
4 points.  It then prints that to file.
//PrintAllData uses PrintData 9 times to do this for the entire grid.

int FillGrid ( RGBImageType::Pointer ImageToDrawOn, vector<int> ArrayOfXIndices,
vector<int> ArrayOfYIndices, vector<int> ListOfQuads, vector<float> AreaChanges );

int FillBox ( RGBImageType::Pointer ImageToDrawOn, int x1, int y1, int x2, int y2, int
x3, int y3, int x4, int y4, vector<float> AreaChanges );

int main()
{
	//First I will make a variable for the input filename, which will be put in at
command line using the cin function
	char* inputFilename = new char[50];

	cout << "Please enter the stack's file name (must be a TIFF file)." << endl;
	cout << "WARNING:: Entering more than 50 characters will cause an error." << endl;
	cout << "Input stack file name: ";
	cin >> inputFilename;

	stringstream inputNames;
	inputNames << inputFilename << ".tif" ;
```

```
      ReaderType3D::Pointer reader = ReaderType3D::New();
                           //creating the reader
    reader->SetFileName( inputNames.str().c_str() );
                    //the reader will now look for the file name input by the user

      try
    {
        reader->Update();
                                                  // reader will attempt to update with
inputted filename
    }
      catch (itk::ExceptionObject &e)
    {
    cerr << "Image not found." << endl << e << endl;
                    // if file cannot be found, program will exit
    return EXIT_FAILURE;
    }

      cout << "File found." << endl;

      int xCenter;
      int yCenter;
      int xLength;
      int yLength;
      int xGridSize;
      int yGridSize;
      cout << "Center X-coordinate: ";
      cin >> xCenter;
      cout << "Center Y-coordinate: ";
      cin >> yCenter;
      cout << "X-length: ";
      cin >> xLength;
      cout << "Y-length: ";
      cin >> yLength;
      cout << "X Grid Size: ";
      cin >> xGridSize;
      cout << "Y Grid Size: ";
      cin >> yGridSize;


      //establishing variables for the input image's largest possible region and size
      ImageType3D::RegionType inputRegion = (reader->GetOutput())-
>GetLargestPossibleRegion();
      ImageType3D::SizeType inputSize = (reader->GetOutput())-
>GetLargestPossibleRegion().GetSize();


      int numCols = xGridSize+1;
      int numRows = yGridSize+1;
      int numPoints=(xGridSize+1)*(yGridSize+1);
      int xBegin = xCenter-(xLength/2);
      int yBegin = yCenter-(yLength/2);
      int xEnd = xBegin + xLength;
      int yEnd = yBegin + yLength;
      int xQuad = (xEnd-xBegin)/xGridSize;
      int yQuad = (yEnd-yBegin)/yGridSize;
      //cout << "numCols = " << numCols << "; numRows = " << numRows << endl;
```

```
	std::vector<int> xCoordinates;
	std::vector<int> yCoordinates;
	std::vector<float> xCoordinatesActual;
	std::vector<float> yCoordinatesActual;

	std::vector<int> listOfQuads;
	int numQuads = xGridSize*yGridSize;

	for(int h=0; h<(numPoints); h++)
	{
		int pointNumber = h+1;
		int rowNumber = floor(h/(numCols))+1;
		int colNumber = pointNumber-((rowNumber-1)*numCols);
		int xCoord = xBegin+(colNumber-1)*xQuad;
		int yCoord = yBegin+(rowNumber-1)*yQuad;
		xCoordinates.push_back(xCoord);
		yCoordinates.push_back(yCoord);
		xCoordinatesActual.push_back(xCoord);
		yCoordinatesActual.push_back(yCoord);
		//cout<< " rowNum = " << rowNumber << "; colNumber = " << colNumber <<
endl;
		//cout<<"Point " << pointNumber << "is at [" << xCoord << ", " << yCoord <<
"]." << endl;
		if((colNumber < (xGridSize+1)) && (rowNumber < (yGridSize+1)))
		{
			int firstPoint = h;
			int secondPoint = h+1;
			int thirdPoint = secondPoint+numCols;
			int fourthPoint = thirdPoint-1;
			cout << firstPoint << " " << secondPoint << " " << thirdPoint << " "
<< fourthPoint << " " << endl;
			listOfQuads.push_back(firstPoint);
			listOfQuads.push_back(secondPoint);
			listOfQuads.push_back(thirdPoint);
			listOfQuads.push_back(fourthPoint);
		}
	}


	for(int g=0; g<numPoints; g++)
	{
		int quadNumber = g;
		int rowNumber = floor(quadNumber/(yGridSize))+1;
		int colNumber = quadNumber-((rowNumber-1)*xGridSize);

	}

	// Setting up the registration filter
	RegistrationType::Pointer registrationFilter = RegistrationType::New();
    registrationFilter->SetMaxLevel(1);
    registrationFilter->SetUseNormalizedGradient( true );
    registrationFilter->ChooseMetric( 0 );

	// Many of the following parameters were not changed from the example
registration.
	// I did change meshpixels around, but decided that 4 was the perfect balance of
speed vs. resolution
	// Number of iterations is 15 -- again to balance between speed and accuracy.
```

```
	// Elasticity and rho, while ITK recommended E/p to be roughly equal to 1, I found
that having a high elasticity and a low rho allowed for more meaningful strain maps.
	int meshPixels;
	meshPixels = 6;                                  // affects size of finite elements
(and pixel resolution of strain maps).  low values EXTREMELY slow

	unsigned int maxiters = 20;
	float        E = 1000;                   //weight parameter for elasticity matrix
	float        p = 1;                      //weight parameter for mass matrix
	float          gamma = 1.5;        //weight parameter for how similar the two
images should be
	registrationFilter->SetElasticity(E, 0);

	registrationFilter->SetRho(p, 0);

	registrationFilter->SetGamma(gamma, 0);

	registrationFilter->SetAlpha(1.);
	registrationFilter->SetMaximumIterations( maxiters, 0 );
	registrationFilter->SetMeshPixelsPerElementAtEachResolution(meshPixels, 0);
	registrationFilter->SetWidthOfMetricRegion(1, 0);
	registrationFilter->SetNumberOfIntegrationPoints(2, 0);
	registrationFilter->SetDoLineSearchOnImageEnergy( 0 );
	registrationFilter->SetTimeStep(1.);
	registrationFilter->SetEmployRegridding(false);
	registrationFilter->SetUseLandmarks(true);

	// Create the material properties
	itk::fem::MaterialLinearElasticity::Pointer m;
	m = itk::fem::MaterialLinearElasticity::New();
	m->SetGlobalNumber(0);
	m->SetYoungsModulus(registrationFilter->GetElasticity()); // Young's modulus of
the membrane
	m->SetCrossSectionalArea(1.0);                            // Cross-sectional area
	m->SetThickness(1.0);                                     // Thickness
	m->SetMomentOfInertia(1.0);                               // Moment of inertia
	m->SetPoissonsRatio(0.5);                                 // Poisson's ratio --
tissues are mostly composed of water, which is incompressible.  So 0.5 makes sense here.
	m->SetDensityHeatProduct(1.0);                            // Density-Heat capacity
product

	// Create the element type
	ElementType::Pointer e1=ElementType::New();
	e1->SetMaterial(m.GetPointer());
	registrationFilter->SetElement(e1.GetPointer());
	registrationFilter->SetMaterial(m);


	//the size of the slices will be identical to the input images, except with a size
of 0 in the z-direction
	ImageType3D::SizeType desiredSize = inputRegion.GetSize();
	desiredSize[2] = 0;


	//this allows the user to choose how many slices to the program on, so you need
not run the entire registration (which takes a long time) if you
	//are debugging or adjusting parameters.  There is also a checkpoint to assure the
range is within the actual size of the image.
```

```
        int range = inputSize[2];
        int difference;
        difference = 1;

        //create a mock data
set~~~
```

```
	//making a variable for the output data file's name, which can be input by the
user.  The file name is then put into a FILE* pointer to create, open and write the data.
	//The first line is written at this point, which contains all the headers of the
data.  The format may look strange (\t indicating tabs), but when opened in Microsoft
	//Excel, it is very well organized.
	stringstream outputNames;
	outputNames << "outputData_" << inputFilename << ".txt" ;
	FILE * pFile2 = fopen ( outputNames.str().c_str(), "w" );
	fprintf(pFile2,
"Registration\tQuadrilateral\tA\tB\tC\tD\tA_x\tA_y\tB_x\tB_y\tC_x\tC_y\tD_x\tD_y\tL:W
ratio\tArea\n");

	std::vector<float> fixedArea;
	std::vector<float> movingArea;

	// Finally the for loop, which will for iterative registrations, is setup.  The
range input by the user is used as the stopping point here.
	for(unsigned int j = 0; j < (range - difference); j++)
	{
		//Cout number j so the user will know how far along their program is
		cout << "Starting Registration #" << j << endl;

		// ExtractFilterType will allow for the extraction of 2D slices from the 3D
stack read in by the reader.  Two extractors are necessary
		// because the registration filter requires two images for a registration.
		// Initializing them now makes sense because these will be constant
throughout the registration loop.
		ExtractFilterType::Pointer extractor1 = ExtractFilterType::New();
		extractor1->InPlaceOn();
		extractor1->SetDirectionCollapseToSubmatrix();
		ExtractFilterType::Pointer extractor2 = ExtractFilterType::New();
		extractor2->InPlaceOn();
		extractor2->SetDirectionCollapseToSubmatrix();
		RGBExtractFilterType::Pointer RGBextractor1 = RGBExtractFilterType::New();
		RGBextractor1->InPlaceOn();
		RGBextractor1->SetDirectionCollapseToSubmatrix();
		RGBExtractFilterType::Pointer RGBextractor2 = RGBExtractFilterType::New();
		RGBextractor2->InPlaceOn();
		RGBextractor2->SetDirectionCollapseToSubmatrix();

		//Extractor1 will get slice j while extractor2 gets slice (j+1).  These can
then be plugged into the registration filter.
		ImageType3D::IndexType extractorStart1 = inputRegion.GetIndex();
		extractorStart1[2] = j;
		ImageType3D::RegionType desiredRegion1;
		desiredRegion1.SetSize( desiredSize );
		desiredRegion1.SetIndex( extractorStart1 );
		extractor1->SetExtractionRegion( desiredRegion1 );
		extractor1->SetInput( reader->GetOutput() );
		RGBextractor1->SetExtractionRegion( desiredRegion1 );
		RGBextractor1->SetInput( reader->GetOutput() );
		RGBextractor1->Update();

		ImageType3D::IndexType desiredStart2 = inputRegion.GetIndex();
		desiredStart2[2] = j + difference;
```

```
            ImageType3D::RegionType desiredRegion2;
            desiredRegion2.SetSize( desiredSize );
            desiredRegion2.SetIndex( desiredStart2 );
            extractor2->SetExtractionRegion( desiredRegion2 );
            extractor2->SetInput( reader->GetOutput() );
            RGBextractor2->SetExtractionRegion( desiredRegion2 );
            RGBextractor2->SetInput( reader->GetOutput() );
            RGBextractor2->Update();

            RescalerType::Pointer rescaler1 = RescalerType::New();
            RescalerType::Pointer rescaler2 = RescalerType::New();
            RescalerType::Pointer maskRescaler = RescalerType::New();
            rescaler1->SetInput(extractor1->GetOutput() );
            rescaler2->SetInput(extractor2->GetOutput() );
            const double desiredMinimum =  0.0;
            const double desiredMaximum =  255.0;
            rescaler1->SetOutputMinimum( desiredMinimum );
            rescaler1->SetOutputMaximum( desiredMaximum );
            rescaler1->UpdateLargestPossibleRegion();
            rescaler2->SetOutputMinimum( desiredMinimum );
            rescaler2->SetOutputMaximum( desiredMaximum );
            rescaler2->UpdateLargestPossibleRegion();

            // The histogram filter helps match the images -- it was taken from the
example registration provided in the ITK documentation
            // Parameters for the historgram filter have not been changed from the
example -- these seemed to work well.
            HEFilterType::Pointer IntensityEqualizeFilter = HEFilterType::New();
            IntensityEqualizeFilter->SetReferenceImage( rescaler1->GetOutput() );
            IntensityEqualizeFilter->SetInput( rescaler2->GetOutput() );
            IntensityEqualizeFilter->SetNumberOfHistogramLevels(100);
            IntensityEqualizeFilter->SetNumberOfMatchPoints(15);
            IntensityEqualizeFilter->ThresholdAtMeanIntensityOn();
            IntensityEqualizeFilter->Update();

            // Setting the images for registration filter and running the registration
            registrationFilter->SetFixedImage( rescaler1->GetOutput() );
            registrationFilter->SetMovingImage(IntensityEqualizeFilter->GetOutput());
            registrationFilter->Modified();
            registrationFilter->RunRegistration();

            // The X- and Y- displacements from the registration need to be written and
stored.  deltaX and deltaY are initialized to do so.  the initial displacement image is
also initialized
            DispWriterType::Pointer dispWriter = DispWriterType::New();
            DispImage::Pointer displacement = DispImage::New();
            displacement = registrationFilter->GetDisplacementField();

            // stringstreams are set up for sequential output file names, using j as
the number to be printed.
            // the WriteImage function was created to make this process easier, as I am
writing multiple images.  See below main function for explanation.

            stringstream gridNames;
            gridNames << "grid_" << inputFilename << "_" << j << ".tif";
            RGBImageType::Pointer fixedImage = RGBImageType::New();
            fixedImage->SetRegions(extractor1->GetOutput()-
>GetLargestPossibleRegion());
```

```
		fixedImage->Allocate();
		fixedImage = RGBextractor1->GetOutput();

		DrawGrid( fixedImage, xCoordinates, yCoordinates, listOfQuads );

		RGBWriterType::Pointer RGBwriter = RGBWriterType::New();
		RGBwriter->SetFileName( gridNames.str().c_str() );
		RGBwriter->SetInput( fixedImage );
		RGBwriter->SetImageIO(tiffIO);
		RGBwriter->Update();

		for(int r=0; r<numPoints; r++)
		{
			xCoordinatesActual.at(r) = UpdateIndices( xCoordinatesActual,
yCoordinatesActual, registrationFilter->GetDisplacementField(), r, 0 );
			yCoordinatesActual.at(r) = UpdateIndices( xCoordinatesActual,
yCoordinatesActual, registrationFilter->GetDisplacementField(), r, 1 );
			xCoordinates.at(r) = Round(xCoordinatesActual.at(r));
			yCoordinates.at(r) = Round(yCoordinatesActual.at(r));
		}

		stringstream grid2Names;
		grid2Names << "grid_" << inputFilename << "_" << (j+1) << ".tif";

		RGBImageType::Pointer movingImage = RGBImageType::New();
		movingImage = RGBextractor2->GetOutput();
		movingImage->Allocate();
		DrawGrid( movingImage, xCoordinates, yCoordinates, listOfQuads );
		RGBwriter->SetFileName( grid2Names.str().c_str() );
		RGBwriter->SetInput( movingImage );
		RGBwriter->Update();

		extractor2->Update();


		movingArea.clear();
		PrintAllData(xCoordinates, yCoordinates, listOfQuads, j, pFile2,
movingArea);

		vector<float> areaChanges;
		for(int e=0; e<numQuads; e++_
		{
			areaChanges.push_back(movingArea.at(e)/fixedArea.at(e));
		}



		fixedArea = movingArea;
	}

	//deallocating memory
	delete [] inputFilename;

	system("PAUSE");
	return EXIT_SUCCESS;

}
```

```
//Functions~~~
```

```
	i3[1]= y3;

	DispIndex i4;
	i4[0]= x4;
	i4[1]= y4;

	ImageType2D::SizeType imageSize = (ImageToDrawOn-
>GetLargestPossibleRegion().GetSize());

	if ( (i1[0] < imageSize[0]) && (i2[0] < imageSize[0]) && (i1[1] < imageSize[1]) &&
(i2[1] < imageSize[1]) )
	{
		itk::LineIterator<RGBImageType> drawline1(ImageToDrawOn, i1, i2);
drawline1.GoToBegin();
		while (!drawline1.IsAtEnd())  {drawline1.Set(red); ++drawline1;}
	}

	if ( (i2[0] < imageSize[0]) && (i3[0] < imageSize[0]) && (i2[1] < imageSize[1]) &&
(i3[1] < imageSize[1]) )
	{
		itk::LineIterator<RGBImageType> drawline2(ImageToDrawOn, i2, i3);
drawline2.GoToBegin();
		while (!drawline2.IsAtEnd())  {drawline2.Set(red); ++drawline2;}
	}
	if ( (i3[0] < imageSize[0]) && (i4[0] < imageSize[0]) && (i3[1] < imageSize[1]) &&
(i4[1] < imageSize[1]) )
	{
		itk::LineIterator<RGBImageType> drawline3(ImageToDrawOn, i3, i4);
drawline3.GoToBegin();
		while (!drawline3.IsAtEnd())  {drawline3.Set(red); ++drawline3;}
	}
	if ( (i1[0] < imageSize[0]) && (i4[0] < imageSize[0]) && (i1[1] < imageSize[1]) &&
(i4[1] < imageSize[1]) )
	{
		itk::LineIterator<RGBImageType> drawline4(ImageToDrawOn, i4, i1);
drawline4.GoToBegin();
		while (!drawline4.IsAtEnd())  {drawline4.Set(red); ++drawline4;}
	}

	return EXIT_SUCCESS;
}
//DrawBox uses 4 ITK line iterators to draw one of the quadrilateral of the distortion
diagram.



int DrawGrid ( RGBImageType::Pointer ImageToDrawOn, vector<int> ArrayOfXIndices,
vector<int> ArrayOfYIndices, vector<int> ListOfQuads )
{
	for(int b = 0; b < ListOfQuads.size(); b+=4)
	{
		DrawBox(ImageToDrawOn, ArrayOfXIndices[ListOfQuads[b]],
ArrayOfYIndices[ListOfQuads[b]],
			ArrayOfXIndices[ListOfQuads[b+1]],
ArrayOfYIndices[ListOfQuads[b+1]],
			ArrayOfXIndices[ListOfQuads[b+2]],
ArrayOfYIndices[ListOfQuads[b+2]],
```

```
                    ArrayOfXIndices[ListOfQuads[b+3]],
ArrayOfYIndices[ListOfQuads[b+3]]);
        }


        return EXIT_SUCCESS;
}
//DrawBox is repeated 9 times, using the color input by the user at the command line and
the array of indices to generate a 3x3 grid to overlay on the image.


void SetupIndices (int PointNumber, DispIndex* ArrayOfIndices, float*
ArrayOfIndicesActual, DispIndex Index, int XCoord, int YCoord )
{
        //assists to put the coordinates into an array of indices
        Index[0] = XCoord;
        Index[1] = YCoord;
        ArrayOfIndices[PointNumber] = Index;

        int actualPointNumber1 = (PointNumber*2);
        int actualPointNumber2 = (PointNumber*2)+1;
        ArrayOfIndicesActual[actualPointNumber1] = XCoord;
        ArrayOfIndicesActual[actualPointNumber2] = YCoord;
}

float UpdateIndices (vector<float> ArrayOfXIndices, vector<float> ArrayOfYIndices,
DispImage::Pointer DisplacementImage, int PointNumber, int XorY)
{
        DispIndex temp;
        temp[0] = Round(ArrayOfXIndices[PointNumber]);
        temp[1] = Round(ArrayOfYIndices[PointNumber]);

        float xDisp = (DisplacementImage->GetPixel(temp))[0];
        float yDisp = (DisplacementImage->GetPixel(temp))[1];
        float newX = ArrayOfXIndices[PointNumber] + xDisp;
        float newY = ArrayOfYIndices[PointNumber] + yDisp;

        if(XorY == 0)
        {
                return newX;
        }
        else
        {
                return newY;
        }
}


void PrintData(vector<int> ArrayOfXIndices, vector<int> ArrayOfYIndices, int
RegistrationNumber, int WhichSquare, int A, int B, int C, int D, FILE* textFilePointer,
vector<float> MovingAreas)
{
        // Using the distance formula to calculate the distance of each of the four sides
of the quadrilateral
        int aX = ArrayOfXIndices[A];
        int bX = ArrayOfXIndices[B];
        int cX = ArrayOfXIndices[C];
        int dX = ArrayOfXIndices[D];
```

```
	int aY = ArrayOfYIndices[A];
	int bY = ArrayOfYIndices[B];
	int cY = ArrayOfYIndices[C];
	int dY = ArrayOfYIndices[D];


	double AB = sqrt ( ((aX-bX)*(aX-bX)) + ((aY-bY)*(aY-bY)) );
	double CB = sqrt ( ((cX-bX)*(cX-bX)) + ((cY-bY)*(cY-bY)) );
	double CD = sqrt ( ((cX-dX)*(cX-dX)) + ((cY-dY)*(cY-dY)) );
	double AD = sqrt ( ((aX-dX)*(aX-dX)) + ((aY-dY)*(aY-dY)) );

	//Each quadrilateral has 4 L:W ratios, so I'm just taking the average of them
	double AvgLengthWidthRatio = ((AB/CB)+(AB/AD)+(CD/CB)+(CD/AD))/4;

	//Using the Brahmagupta's formula for a CYCLIC quadrilateral
	// Area = sqrt( (s-AB)(s-BC)(s-CD)(s-DA) ) where s is the semi-perimeter (half the
perimeter)

	double S = ((AB+CB+CD+AD)/2);

	//NOTE::::: Points A-D must be ordered in a CLOCKWISE fashion around the
quadrilateral for area to calculate correctly.
	double Area = sqrt( (AB-S)*(CB-S)*(CD-S)*(AD-S) );

	MovingAreas.push_back(Area);

	//Printing data
	//While the format may look weird, it is organized properly when opened in
Microsoft Excel.
	fprintf(textFilePointer,
"%i\t%i\t%i\t%i\t%i\t%i\t%i\t%i\t%i\t%i\t%i\t%i\t%i\t%i\t%f\t%f\n", RegistrationNumber,
WhichSquare, A, B, C, D, aX, aY,
		bX, bY, cX, cY, dX, dY, AvgLengthWidthRatio, Area);

}

void PrintAllData( vector<int> ArrayOfXIndices, vector<int> ArrayOfYIndices, vector<int>
ListOfQuads, int RegistrationNumber, FILE* textFilePointer, vector<float> MovingAreas )
{
	for(int a = 0; a < ListOfQuads.size(); a+=4)
	{
		int quadNumber = (a/4)+1;
		int firstPoint = ListOfQuads[a];
		int secondPoint = ListOfQuads[a+1];
		int thirdPoint = ListOfQuads[a+2];
		int fourthPoint = ListOfQuads[a+3];
		PrintData(ArrayOfXIndices, ArrayOfYIndices, RegistrationNumber, quadNumber,
firstPoint, secondPoint, thirdPoint, fourthPoint, textFilePointer, MovingAreas);
	}
}
```

B.3 YAP NUCLEAR LOCALIZATION ANALYSIS

The microsurgical preparation to create mesenchymal cell aggregates was discussed in a previous section (Section 4.4.1). For this analysis, embryos were first injected with mRNA encoding ZO-1-RFP and YAP-GFP. After creating aggregates, they were then fixed and stained with Hoescht 33342 (1:2000; 7 minutes).

Once imaged and z-projected, the nuclear channel was used to manually create a binary nuclear mask using auto-thresholding in ImageJ. Once the binary mask image is created, you can do the following:

1.) Use the "Edit → Selection →Create Selection" function to generate one ROI for the entire nuclei mask

2.) Add it to the ROI manager, highlight it and then go to "More → Split." This will create one ROI for each nucleus in the mask.

3.) Open the YAP channel and name the image "yap".

4.) Run the macro code. It will iterate over each nucleus ROI, measure the mean YAP intensity for that ROI, then create a "cytoplasmic" ROI by dilating and subtracting the original nuclear region then final measure the cytoplasmic YAP intensity (See Fig. 27).

The following code is a macro written for ImageJ:

```
newImage("cytoYAP", "32-bit black", getWidth(), getHeight(), 1);
newImage("nucYAP", "32-bit black", getWidth(), getHeight(), 1);

for (i=0 ; i<(roiManager("count")); i++) {

    selectImage("yap");
  roiManager("select", i);
    getStatistics(area, mean1);
    setColor(mean1);

    selectImage("nucYAP");
  roiManager("select", i);
    fill();

    run("Create Mask");
    run("Duplicate...", " ");
    setOption("BlackBackground", true);
    run("Dilate");
    run("Dilate");
    run("Dilate");
    imageCalculator("Subtract create", "Mask-1","Mask");
    selectWindow("Result of Mask-1");
    run("Create Selection");
    run("Make Inverse");
```

```
        roiManager("Add");

        selectWindow("Mask");
        close();
        selectWindow("Mask-1");
        close();
        selectWindow("Result of Mask-1");
        close();

        selectWindow("yap");

        var1 = roiManager("count");
        var2 = var1 - 1;
        roiManager("select", var2);
    getStatistics(area, mean);
        setColor(mean);

        selectWindow("cytoYAP");

        roiManager("select", i);
        fill();

        roiManager("select", var2);
        roiManager("Delete");
}
```

BIBLIOGRAPHY

Abu-Issa, R. (2014). "Heart fields: spatial polarity and temporal dynamics." Anatomical record (Hoboken, N.J. : 2007) **297**: 175-182.

Abu-Issa, R. and M. L. Kirby (2007). "Heart field: from mesoderm to heart tube." Annu Rev Cell Dev Biol **23**: 45-68.

Abualsaud, A. O., B. S. Lowe, K. Guo, A. J. Marelli, M. Kaouache, L. Guo, L. Jutras, G. Martucci and J. Therrien (2016). "Cardiac output as a predictor in congenital heart disease: Are we stating the obvious?" Int J Cardiol **210**: 143-148.

Adams, C. L., Y.-T. Chen, S. J. Smith and W. James Nelson (1998). "Mechanisms of Epithelial Cell–Cell Adhesion and Cell Compaction Revealed by High-resolution Tracking of E-Cadherin– Green Fluorescent Protein." The Journal of Cell Biology **142**(4): 1105-1119.

Aleksandrova, A., A. Czirok, E. Kosa, O. Galkin, T. J. Cheuvront and B. J. Rongish (2015). "The endoderm and myocardium join forces to drive early heart tube assembly." Developmental Biology **404**: 40-54.

Aleksandrova, A., A. Czirók, A. Szabó, M. B. Filla, M. J. Hossain, P. F. Whelan, R. Lansford and B. J. Rongish (2012). "Convective tissue movements play a major role in avian endocardial morphogenesis." Developmental biology **363**: 348-361.

Andrew, D. J. and A. J. Ewald (2010). "Morphogenesis of epithelial tubes: Insights into tube formation, elongation, and elaboration." Dev Biol **341**(1): 34-55.

Aragona, M., T. Panciera, A. Manfrin, S. Giulitti, F. Michielin, N. Elvassore, S. Dupont and S. Piccolo (2013). "A mechanical checkpoint controls multicellular growth through YAP/TAZ regulation by actin-processing factors." Cell **154**(5): 1047-1059.

Ariizumi, T. and M. Asashima (2001). "In vitro induction systems for analyses of amphibian organogenesis and body patterning." Int J Dev Biol **45**(1): 273-279.

Atsuta, Y. and Y. Takahashi (2015). "FGF8 coordinates tissue elongation and cell epithelialization during early kidney tubulogenesis." Development **142**(13): 2329-2337.

Attia, L., J. Schneider, R. Yelin and T. M. Schultheiss (2015). "Collective cell migration of the nephric duct requires FGF signaling." Dev Dyn **244**(2): 157-167.

Austen, K., P. Ringer, A. Mehlich, A. Chrostek-Grashoff, C. Kluger, C. Klingner, B. Sabass, R. Zent, M. Rief and C. Grashoff (2015). "Extracellular rigidity sensing by talin isoform-specific mechanical linkages." Nature cell biology **17**(12): 1597-1606.

Bai, H., Q. Zhu, A. Surcel, T. Luo, Y. Ren, B. Guan, Y. Liu, N. Wu, N. E. Joseph, T. L. Wang, N. Zhang, D. Pan, G. Alpini, D. N. Robinson and R. A. Anders (2016). "Yes-associated protein impacts adherens junction assembly through regulating actin cytoskeleton organization." Am J Physiol Gastrointest Liver Physiol **311**(3): G396-411.

Barkan, D., L. H. El Touny, A. M. Michalowski, J. A. Smith, I. Chu, A. S. Davis, J. D. Webster, S. Hoover, R. M. Simpson, J. Gauldie and J. E. Green (2010). "Metastatic growth from dormant cells induced by a col-I-enriched fibrotic environment." Cancer Res **70**(14): 5706-5716.

Barkan, D., L. H. El Touny, A. M. Michalowski, J. A. Smith, I. Chu, A. S. Davis, J. D. Webster, S. Hoover, R. M. Simpson, J. Gauldie and J. E. Green (2010). "Metastatic Growth from Dormant Cells Induced by a Col-I–Enriched Fibrotic Environment." Cancer Research **70**(14): 5706-5716.

Barkan, D., H. Kleinman, J. L. Simmons, H. Asmussen, A. K. Kamaraju, M. J. Hoenorhoff, Z.-y. Liu, S. V. Costes, E. H. Cho, S. Lockett, C. Khanna, A. F. Chambers and J. E. Green (2008). "Inhibition of Metastatic Outgrowth from Single Dormant Tumor Cells by Targeting the Cytoskeleton." Cancer Research **68**(15): 6241-6250.

Barrios, A., R. J. Poole, L. Durbin, C. Brennan, N. Holder and S. W. Wilson (2003). "Eph/Ephrin signaling regulates the mesenchymal-to-epithelial transition of the paraxial mesoderm during somite morphogenesis." Current Biology **13**(18): 1571-1582.

Bartman, T. and J. Hove (2005). "Mechanics and function in heart morphogenesis." Dev Dyn **233**(2): 373-381.

Battle, C., C. M. Ott, D. T. Burnette, J. Lippincott-Schwartz and C. F. Schmidt (2015). "Intracellular and extracellular forces drive primary cilia movement." Proc Natl Acad Sci U S A **112**(5): 1410-1415.

Bauer, D. V., S. Huang and S. A. Moody (1994). "The cleavage stage origin of Spemann's Organizer: analysis of the movements of blastomere clones before and during gastrulation in Xenopus." Development **120**(5): 1179-1189.

Bayraktar, M. and J. Manner (2014). "Cardiac looping may be driven by compressive loads resulting from unequal growth of the heart and pericardial cavity. Observations on a physical simulation model." Front Physiol **5**: 112.

Benazeraf, B., P. Francois, R. E. Baker, N. Denans, C. D. Little and O. Pourquie (2010). "A random cell motility gradient downstream of FGF controls elongation of an amniote embryo." Nature **466**(7303): 248-252.

Bhumiratana, S., R. E. Eton, S. R. Oungoulian, L. Q. Wan, G. A. Ateshian and G. Vunjak-Novakovic (2014). "Large, stratified, and mechanically functional human cartilage grown in vitro by mesenchymal condensation." Proceedings of the National Academy of Sciences **111**(19): 6940-6945.

Bogatan, S., D. Cevik, V. Demidov, J. Vanderploeg, A. Panchbhaya, A. Vitkin and J. R. Jacobs (2015). "Talin Is Required Continuously for Cardiomyocyte Remodeling during Heart Growth in Drosophila." PLoS One **10**(6): e0131238.

Brade, T., S. Gessert, M. Kuhl and P. Pandur (2007). "The amphibian second heart field: Xenopus islet-1 is required for cardiovascular development." Dev Biol **311**(2): 297-310.

Brueckner, M. (2007). "Heterotaxia, congenital heart disease, and primary ciliary dyskinesia." Circulation **115**(22): 2793-2795.

Burute, M., M. Prioux, G. Blin, S. Truchet, G. Letort, Q. Tseng, T. Bessy, S. Lowell, J. Young, O. Filhol and M. Thery (2017). "Polarity Reversal by Centrosome Repositioning Primes Cell Scattering during Epithelial-to-Mesenchymal Transition." Dev Cell **40**(2): 168-184.

Campàs, O. (2016). A toolbox to explore the mechanics of living embryonic tissues. Seminars in cell & developmental biology, Elsevier.

Campbell, K. and J. Casanova (2015). "A role for E-cadherin in ensuring cohesive migration of a heterogeneous population of non-epithelial cells." Nat Commun **6**: 7998.

Carver, E. A., R. Jiang, Y. Lan, K. F. Oram and T. Gridley (2001). "The mouse snail gene encodes a key regulator of the epithelial-mesenchymal transition." Mol Cell Biol **21**(23): 8184-8188.

Chaffer, C. L., J. P. Brennan, J. L. Slavin, T. Blick, E. W. Thompson and E. D. Williams (2006). "Mesenchymal-to-Epithelial Transition Facilitates Bladder Cancer Metastasis: Role of Fibroblast Growth Factor Receptor-2." Cancer Research **66**(23): 11271-11278.

Chaffer, C. L., E. W. Thompson and E. D. Williams (2007). "Mesenchymal to epithelial transition in development and disease." Cells Tissues Organs **185**(1-3): 7-19.

Chalmers, A. D., K. Lachani, Y. Shin, V. Sherwood, K. W. Cho and N. Papalopulu (2006). "Grainyhead-like 3, a transcription factor identified in a microarray screen, promotes the specification of the superficial layer of the embryonic epidermis." Mech Dev **123**(9): 702-718.

Chalmers, A. D., M. Pambos, J. Mason, S. Lang, C. Wylie and N. Papalopulu (2005). "aPKC, Crumbs3 and Lgl2 control apicobasal polarity in early vertebrate development." Development **132**(5): 977-986.

Chalmers, A. D., B. Strauss and N. Papalopulu (2003). "Oriented cell divisions asymmetrically segregate aPKC and generate cell fate diversity in the early Xenopus embryo." Development **130**(12): 2657-2668.

Chalmers, A. D., D. Welchman and N. Papalopulu (2002). "Intrinsic differences between the superficial and deep layers of the Xenopus ectoderm control primary neuronal differentiation." Dev Cell **2**(2): 171-182.

Chanet, S. and A. C. Martin (2014). "Mechanical force sensing in tissues." Progress in molecular biology and translational science **126**: 317.

Chao, Y., Q. Wu, M. Acquafondata, R. Dhir and A. Wells (2012). "Partial Mesenchymal to Epithelial Reverting Transition in Breast and Prostate Cancer Metastases." Cancer Microenvironment **5**(1): 19-28.

Chartier, A., S. Zaffran, M. Astier, M. Sémériva and D. Gratecos (2002). "Pericardin, a Drosophila type IV collagen-like protein is involved in the morphogenesis and maintenance of the heart epithelium during dorsal ectoderm closure." Development **129**(13): 3241-3253.

Chen, X., Y. An, Y. Gao, L. Guo, L. Rui, H. Xie, M. Sun, S. Lam Hung, X. Sheng, J. Zou, Y. Bao, H. Guan, B. Niu, Z. Li, R. H. Finnell, J. F. Gusella, B. L. Wu and T. Zhang (2016). "Rare Deleterious PARD3 Variants in the aPKC-Binding Region are Implicated in the Pathogenesis of Human Cranial Neural Tube Defects Via Disrupting Apical Tight Junction Formation." Hum Mutat.

Chen, X. and I. G. Macara (2005). "Par-3 controls tight junction assembly through the Rac exchange factor Tiam1." Nat Cell Biol **7**(3): 262-269.

Chien, Y.-H., R. Keller, C. Kintner and D. R. Shook (2015). "Mechanical Strain Determines the Axis of Planar Polarity in Ciliated Epithelia." Current Biology.

Chien, Y. H., R. Keller, C. Kintner and D. R. Shook (2015). "Mechanical strain determines the axis of planar polarity in ciliated epithelia." Curr Biol **25**(21): 2774-2784.

Choi, B., K. S. Park, J. H. Kim, K. W. Ko, J. S. Kim, D. K. Han and S. H. Lee (2015). "Stiffness of Hydrogels Regulates Cellular Reprogramming Efficiency Through Mesenchymal-to-Epithelial Transition and Stemness Markers." Macromolecular bioscience.

Conway, D. E., M. T. Breckenridge, E. Hinde, E. Gratton, C. S. Chen and M. A. Schwartz (2013). "Fluid shear stress on endothelial cells modulates mechanical tension across VE-cadherin and PECAM-1." Curr Biol **23**(11): 1024-1030.

Cost, A.-L., P. Ringer, A. Chrostek-Grashoff and C. Grashoff (2015). "How to Measure Molecular Forces in Cells: A Guide to Evaluating Genetically-Encoded FRET-Based Tension Sensors." Cellular and molecular bioengineering **8**(1): 96-105.

Costantini, F. and R. Kopan (2010). "Patterning a Complex Organ: Branching Morphogenesis and Nephron Segmentation in Kidney Development." Developmental Cell **18**(5): 698-712.

Cox, T. R., D. Bird, A.-M. Baker, H. E. Barker, M. W.-Y. Ho, G. Lang and J. T. Erler (2013). "LOX-Mediated Collagen Crosslinking Is Responsible for Fibrosis-Enhanced Metastasis." Cancer Research **73**(6): 1721-1732.

Daggett, D. F., C. R. Domingo, P. D. Currie and S. L. Amacher (2007). "Control of morphogenetic cell movements in the early zebrafish myotome." Dev Biol **309**(2): 169-179.

Dale, L. and J. M. Slack (1987). "Fate map for the 32-cell stage of Xenopus laevis." Development (Cambridge, England) **99**: 527-551.

Davidson, L. A., B. D. Dzamba, R. Keller and D. W. Desimone (2008). "Live imaging of cell protrusive activity, and extracellular matrix assembly and remodeling during morphogenesis in the frog, Xenopus laevis." Dev Dyn **237**(10): 2684-2692.

Davidson, L. A., R. Keller and D. W. Desimone (2004). "Assembly and remodeling of the fibrillar fibronectin extracellular matrix during gastrulation and neurulation in *Xenopus laevis*." Dev Dyn **231**(4): 888-895.

Davidson, L. A., M. Marsden, R. Keller and D. W. Desimone (2006). "Integrin alpha5beta1 and fibronectin regulate polarized cell protrusions required for Xenopus convergence and extension." Curr Biol **16**(9): 833-844.

Davidson, L. A., G. F. Oster, R. E. Keller and M. A. Koehl (1999). "Measurements of mechanical properties of the blastula wall reveal which hypothesized mechanisms of primary invagination are physically plausible in the sea urchin Strongylocentrotus purpuratus." Dev Biol **209**(2): 221-238.

De Vries, W. N., A. V. Evsikov, B. E. Haac, K. S. Fancher, A. E. Holbrook, R. Kemler, D. Solter and B. B. Knowles (2004). "Maternal beta-catenin and E-cadherin in mouse development." Development **131**(18): 4435-4445.

Deblandre, G. A., D. A. Wettstein, N. Koyano-Nakagawa and C. Kintner (1999). "A two-step mechanism generates the spacing pattern of the ciliated cells in the skin of Xenopus embryos." Development **126**(21): 4715-4728.

Deniz, E., S. Jonas, M. Khokha and M. A. Choma (2012). "Endogenous contrast blood flow imaging in embryonic hearts using hemoglobin contrast subtraction angiography." Opt Lett **37**(14): 2979-2981.

Desai, L. P., K. E. Chapman and C. M. Waters (2008). "Mechanical stretch decreases migration of alveolar epithelial cells through mechanisms involving Rac1 and Tiam1." Am J Physiol Lung Cell Mol Physiol **295**(5): L958-965.

Discher, D. E., P. Janmey and Y. L. Wang (2005). "Tissue cells feel and respond to the stiffness of their substrate." Science **310**(5751): 1139-1143.

Downing, T. L., J. Soto, C. Morez, T. Houssin, A. Fritz, F. Yuan, J. Chu, S. Patel, D. V. Schaffer and S. Li (2013). "Biophysical regulation of epigenetic state and cell reprogramming." Nat Mater **12**(12): 1154-1162.

Dressler, G. (2002). "Tubulogenesis in the developing mammalian kidney." Trends Cell Biol **12**(8): 390-395.

Dressler, G. R. (2002). "Tubulogenesis in the developing mammalian kidney." Trends in Cell Biology **12**(8): 390-395.

Dubaissi, E., K. Rousseau, R. Lea, X. Soto, S. Nardeosingh, A. Schweickert, E. Amaya, D. J. Thornton and N. Papalopulu (2014). "A secretory cell type develops alongside multiciliated cells, ionocytes and goblet cells, and provides a protective, anti-infective function in the frog embryonic mucociliary epidermis." Development **141**(7): 1514-1525.

Duband, J. L., S. Dufour, K. Hatta, M. Takeichi, G. M. Edelman and J. P. Thiery (1987). "Adhesion molecules during somitogenesis in the avian embryo." J Cell Biol **104**(5): 1361-1374.

Duncan, A. R. and M. K. Khokha (2016). "Xenopus as a model organism for birth defects-Congenital heart disease and heterotaxy." Semin Cell Dev Biol **51**: 73-79.

Dupont, S., L. Morsut, M. Aragona, E. Enzo, S. Giulitti, M. Cordenonsi, F. Zanconato, J. Le Digabel, M. Forcato, S. Bicciato, N. Elvassore and S. Piccolo (2011). "Role of YAP/TAZ in mechanotransduction." Nature **474**(7350): 179-183.

Dykxhoorn, D. M., Y. Wu, H. Xie, F. Yu, A. Lal, F. Petrocca, D. Martinvalet, E. Song, B. Lim and J. Lieberman (2009). "miR-200 enhances mouse breast cancer cell colonization to form distant metastases." PLoS One **4**(9): e7181.

Dzamba, B. J., K. R. Jakab, M. Marsden, M. A. Schwartz and D. W. DeSimone (2009). "Cadherin adhesion, tissue tension, and noncanonical Wnt signaling regulate fibronectin matrix organization." Dev Cell **16**(3): 421-432.

Edlund, A. F., L. A. Davidson and R. E. Keller (2013). "Cell segregation, mixing, and tissue pattern in the spinal cord of the Xenopus laevis neurula." Dev Dyn **242**(10): 1134-1146.

Edouard, T., J. P. Combier, A. Nedelec, S. Bel-Vialar, M. Metrich, F. Conte-Auriol, S. Lyonnet, B. Parfait, M. Tauber, J. P. Salles, F. Lezoualc'h, A. Yart and P. Raynal (2010). "Functional effects of PTPN11 (SHP2) mutations causing LEOPARD syndrome on epidermal growth factor-induced phosphoinositide 3-kinase/AKT/glycogen synthase kinase 3beta signaling." Mol Cell Biol **30**(10): 2498-2507.

Ellawindy, A., K. Satoh, S. Sunamura, N. Kikuchi, K. Suzuki, T. Minami, S. Ikeda, S. Tanaka, T. Shimizu, B. Enkhjargal, S. Miyata, Y. Taguchi, T. Handoh, K. Kobayashi, K. Kobayashi, K. Nakayama, M. Miura and H. Shimokawa (2015). "Rho-Kinase Inhibition During Early Cardiac Development Causes Arrhythmogenic Right Ventricular Cardiomyopathy in Mice." Arterioscler Thromb Vasc Biol **35**(10): 2172-2184.

Engler, A. J., S. Sen, H. L. Sweeney and D. E. Discher (2006). "Matrix elasticity directs stem cell lineage specification." Cell **126**(4): 677-689.

Engler, A. J., S. Sen, H. L. Sweeney and D. E. Discher (2006). "Matrix elasticity directs stem cell lineage specification." Cell **126**(4): 677-689.

Evans, S. M., D. Yelon, F. L. Conlon and M. L. Kirby (2010). "Myocardial lineage development." Circ Res **107**(12): 1428-1444.

Ewald, A. J., A. Brenot, M. Duong, B. S. Chan and Z. Werb (2008). "Collective epithelial migration and cell rearrangements drive mammary branching morphogenesis." Dev Cell **14**(4): 570-581.

Fagotto, F. and B. M. Gumbiner (1994). "Beta-catenin localization during Xenopus embryogenesis: accumulation at tissue and somite boundaries." Development **120(12)**: 3667-3679.

Feroze, R., J. H. Shawky, M. von Dassow and L. A. Davidson (2015). "Mechanics of blastopore closure during amphibian gastrulation." Dev Biol **398**(1): 57-67.

Fesenko, I., T. Kurth, B. Sheth, T. P. Fleming, S. Citi and P. Hausen (2000). "Tight junction biogenesis in the early Xenopus embryo." Mech Dev **96**(1): 51-65.

Fierro-Gonzalez, J. C., M. D. White, J. C. Silva and N. Plachta (2013). "Cadherin-dependent filopodia control preimplantation embryo compaction." Nat Cell Biol **15**(12): 1424-1433.

Fischer, E., E. Legue, A. Doyen, F. Nato, J. F. Nicolas, V. Torres, M. Yaniv and M. Pontoglio (2006). "Defective planar cell polarity in polycystic kidney disease." Nat Genet **38**(1): 21-23.

Fishman, M. C. and K. R. Chien (1997). "Fashioning the vertebrate heart: earliest embryonic decisions." Development **124**(11): 2099-2117.

Fleming, T. P., J. McConnell, M. H. Johnson and B. R. Stevenson (1989). "Development of tight junctions de novo in the mouse early embryo: control of assembly of the tight junction-specific protein, ZO-1." J Cell Biol **108**(4): 1407-1418.

Francou, A., E. Saint-Michel, K. Mesbah and R. G. Kelly (2014). "TBX1 regulates epithelial polarity and dynamic basal filopodia in the second heart field." Development **141**(22): 4320-4331.

Francou, A., E. Saint-Michel, K. Mesbah and R. G. Kelly (2014). "TBX1 regulates epithelial polarity and dynamic basal filopodia in the second heart field." Development **141**: 4320-4331.

Friedl, P. and D. Gilmour (2009). "Collective cell migration in morphogenesis, regeneration and cancer." Nature reviews. Molecular cell biology **10**: 445-457.

Friedland, J. C., M. H. Lee and D. Boettiger (2009). "Mechanically activated integrin switch controls alpha5beta1 function." Science **323**(5914): 642-644.

Fukui, H., K. Terai, H. Nakajima, A. Chiba, S. Fukuhara and N. Mochizuki (2014). S1P-Yap1 Signaling Regulates Endoderm Formation Required for Cardiac Precursor Cell Migration in Zebrafish. Developmental Cell. **31:** 128-136.

Gal, A., T. Sjoblom, L. Fedorova, S. Imreh, H. Beug and A. Moustakas (2008). "Sustained TGF beta exposure suppresses Smad and non-Smad signalling in mammary epithelial cells, leading to EMT and inhibition of growth arrest and apoptosis." Oncogene **27**(9): 1218-1230.

Garavito-Aguilar, Z. V., H. E. Riley and D. Yelon (2010). "Hand2 ensures an appropriate environment for cardiac fusion by limiting Fibronectin function." Development **137**(19): 3215-3220.

Garavito-Aguilar, Z. V., H. E. Riley and D. Yelon (2010). "Hand2 ensures an appropriate environment for cardiac fusion by limiting Fibronectin function." Development (Cambridge, England) **137**: 3215-3220.

Geiger, B., J. P. Spatz and A. D. Bershadsky (2009). "Environmental sensing through focal adhesions." Nat Rev Mol Cell Biol **10**(1): 21-33.

Gerlach, G. F. and R. A. Wingert (2014). "Zebrafish pronephros tubulogenesis and epithelial identity maintenance are reliant on the polarity proteins Prkc iota and zeta." Dev Biol **396**(2): 183-200.

Gessert, S. and M. Kuhl (2009). "Comparative gene expression analysis and fate mapping studies suggest an early segregation of cardiogenic lineages in Xenopus laevis." Dev Biol **334**(2): 395-408.

Ghajar, C. M., H. Peinado, H. Mori, I. R. Matei, K. J. Evason, H. Brazier, D. Almeida, A. Koller, K. A. Hajjar, D. Y. R. Stainier, E. I. Chen, D. Lyden and M. J. Bissell (2013). "The perivascular niche regulates breast tumour dormancy." Nat Cell Biol **15**(7): 807-817.

Gjorevski, N., E. Boghaert and C. M. Nelson (2012). "Regulation of Epithelial-Mesenchymal Transition by Transmission of Mechanical Stress through Epithelial Tissues." Cancer Microenviron **5**(1): 29-38.

Gjorevski, N., E. Boghaert and C. M. Nelson (2012). "Regulation of Epithelial-Mesenchymal Transition by Transmission of Mechanical Stress through Epithelial Tissues." Cancer Microenvironment **5**(1): 29-38.

Gopalakrishnan, S., N. Raman, S. J. Atkinson and J. A. Marrs (1998). "Rho GTPase signaling regulates tight junction assembly and protects tight junctions during ATP depletion." Am J Physiol **275**(3 Pt 1): C798-809.

Gottardi, C. J., M. Arpin, A. S. Fanning and D. Louvard (1996). "The junction-associated protein, zonula occludens-1, localizes to the nucleus before the maturation and during the remodeling of cell-cell contacts." Proc Natl Acad Sci U S A **93**(20): 10779-10784.

Grana, T. M., E. A. Cox, A. M. Lynch and J. Hardin (2010). "SAX-7/L1CAM and HMR-1/cadherin function redundantly in blastomere compaction and non-muscle myosin accumulation during Caenorhabditis elegans gastrulation." Dev Biol **344**(2): 731-744.

Granados-Riveron, J. T. and J. D. Brook (2012). "The impact of mechanical forces in heart morphogenesis." Circ Cardiovasc Genet **5**(1): 132-142.

Grashoff, C., B. D. Hoffman, M. D. Brenner, R. Zhou, M. Parsons, M. T. Yang, M. A. McLean, S. G. Sligar, C. S. Chen, T. Ha and M. A. Schwartz (2010). "Measuring mechanical tension across vinculin reveals regulation of focal adhesion dynamics." Nature **466**(7303): 263-266.

Guilluy, C., V. Swaminathan, R. Garcia-Mata, E. T. O'Brien, R. Superfine and K. Burridge (2011). "The Rho GEFs LARG and GEF-H1 regulate the mechanical response to force on integrins." Nat Cell Biol **13**(6): 722-727.

Gumbiner, B. M. (1996). "Cell adhesion: the molecular basis of tissue architecture and morphogenesis." Cell **84**(3): 345-357.

Gunasinghe, N. P., A. Wells, E. W. Thompson and H. J. Hugo (2012). "Mesenchymal-epithelial transition (MET) as a mechanism for metastatic colonisation in breast cancer." Cancer Metastasis Rev **31**(3-4): 469-478.

Gunasinghe, N. P. A. D., A. Wells, E. W. Thompson and H. J. Hugo (2012). "Mesenchymal–epithelial transition (MET) as a mechanism for metastatic colonisation in breast cancer." Cancer and Metastasis Reviews **31**(3-4): 469-478.

Haack, T., M. Schneider, B. Schwendele and A. D. Renault (2014). "Drosophila heart cell movement to the midline occurs through both cell autonomous migration and dorsal closure." Developmental biology **396**(2): 169-182.

Hamm, M. J., B. C. Kirchmaier and W. Herzog (2016). "Sema3d controls collective endothelial cell migration by distinct mechanisms via Nrp1 and PlxnD1." J Cell Biol **215**(3): 415-430.

Harpaz, N., E. Ordan, K. Ocorr, R. Bodmer and T. Volk (2013). "Multiplexin promotes heart but not aorta morphogenesis by polarized enhancement of slit/robo activity at the heart lumen." Plos Genetics **9**(6): e1003597.

Harris, A. K. (1976). "Is Cell sorting caused by differences in the work of intercellular adhesion? A critique of the Steinberg hypothesis." J Theoretical Biol **61(2)**: 267-285.

Harvey, R. P. (2002). "Patterning the vertebrate heart." Nat Rev Genet **3**(7): 544-556.

Hausen, P. and M. Riebesell (1991). The early development of Xenopus laevis : an atlas of the histology. New York, Springer Verlag.

Heisenberg, C. P. and Y. Bellaiche (2013). "Forces in tissue morphogenesis and patterning." Cell **153**(5): 948-962.

Hempel, A. and M. Kuhl (2016). "A Matter of the Heart: The African Clawed Frog Xenopus as a Model for Studying Vertebrate Cardiogenesis and Congenital Heart Defects." J Cardiovasc Dev Dis **3**(2): 21.

Hierck, B. P., K. Van der Heiden, F. E. Alkemade, S. Van de Pas, J. V. Van Thienen, B. C. Groenendijk, W. H. Bax, A. Van der Laarse, M. C. Deruiter, A. J. Horrevoets and R. E. Poelmann (2008). "Primary cilia sensitize endothelial cells for fluid shear stress." Dev Dyn **237**(3): 725-735.

Hierck, B. P., K. Van der Heiden, C. Poelma, J. Westerweel and R. E. Poelmann (2008). "Fluid shear stress and inner curvature remodeling of the embryonic heart. Choosing the right lane!" ScientificWorldJournal **8**: 212-222.

Hirota, A., K. Kamino, H. Komuro and T. Sakai (1987). "Mapping of early development of electrical activity in the embryonic chick heart using multiple-site optical recording." J Physiol **383**: 711-728.

Hiscox, S., W. G. Jiang, K. Obermeier, K. Taylor, L. Morgan, R. Burmi, D. Barrow and R. I. Nicholson (2006). "Tamoxifen resistance in MCF7 cells promotes EMT-like behaviour and involves modulation of beta-catenin phosphorylation." Int J Cancer **118**(2): 290-301.

Hiscox, S., L. Morgan, T. P. Green, D. Barrow, J. Gee and R. I. Nicholson (2006). "Elevated Src activity promotes cellular invasion and motility in tamoxifen resistant breast cancer cells." Breast Cancer Res Treat **97**(3): 263-274.

Hoffman, J. I. (1995). "Incidence of congenital heart disease: II. Prenatal incidence." Pediatr Cardiol **16**(4): 155-165.

Hoffman, J. I. and S. Kaplan (2002). "The incidence of congenital heart disease." J Am Coll Cardiol **39**(12): 1890-1900.

Hove, J. R., R. W. Koster, A. S. Forouhar, G. Acevedo-Bolton, S. E. Fraser and M. Gharib (2003). "Intracardiac fluid forces are an essential epigenetic factor for embryonic cardiogenesis." Nature **421**(6919): 172-177.

Hundley, W. G., H. F. Li, J. E. Willard, C. Landau, R. A. Lange, B. M. Meshack, L. D. Hillis and R. M. Peshock (1995). "Magnetic resonance imaging assessment of the severity of mitral regurgitation. Comparison with invasive techniques." Circulation **92**(5): 1151-1158.

Inagaki, K., T. Noguchi, T. Matozaki, T. Horikawa, K. Fukunaga, M. Tsuda, M. Ichihashi and M. Kasuga (2000). "Roles for the protein tyrosine phosphatase SHP-2 in cytoskeletal

organization, cell adhesion and cell migration revealed by overexpression of a dominant negative mutant." Oncogene **19**(1): 75-84.

Ishihara, H., B. L. Martin, D. L. Brautigan, H. Karaki, H. Ozaki, Y. Kato, N. Fusetani, S. Watabe, K. Hashimoto, D. Uemura and et al. (1989). "Calyculin A and okadaic acid: inhibitors of protein phosphatase activity." Biochem Biophys Res Commun **159**(3): 871-877.

Ishizaki, T., M. Uehata, I. Tamechika, J. Keel, K. Nonomura, M. Maekawa and S. Narumiya (2000). "Pharmacological properties of Y-27632, a specific inhibitor of rho-associated kinases." Mol Pharmacol **57**(5): 976-983.

Jackson, T. R., H. Y. Kim, U. L. Balakrishnan, C. Stuckenholz and L. A. Davidson (In review). "Proper heart development and function requires temporally controlled mechanical cues to induce mesenchymal-to-epithelial transition in progenitor cells." Curr Biol.

Jessen, J. R., J. Topczewski, S. Bingham, D. S. Sepich, F. Marlow, A. Chandrasekhar and L. Solnica-Krezel (2002). "Zebrafish trilobite identifies new roles for Strabismus in gastrulation and neuronal movements." Nat Cell Biol **4**(8): 610-615.

Jodoin, J. N. and A. C. Martin (2016). "Epithelial Contractility: A Crowning Achievement." Dev Cell **37**(1): 3-4.

Jones, E. and H. Woodland (1987). "The development of animal cap cells in Xenopus: the effects of environment on the differentiation and the migration of grafted ectodermal cells." Development **101**(1): 23-32.

Joos, T. O., C. A. Whittaker, F. Meng, D. W. DeSimone, V. Gnau and P. Hausen (1995). "Integrin alpha 5 during early development of Xenopus laevis." Mech Dev **50**(2-3): 187-199.

Joshi, S. D., M. von Dassow and L. A. Davidson (2010). "Experimental control of excitable embryonic tissues: three stimuli induce rapid epithelial contraction." Exp Cell Res **316**(1): 103-114.

Kang, Y. and K. Pantel (2013). "Tumor cell dissemination: emerging biological insights from animal models and cancer patients." Cancer Cell **23**(5): 573-581.

Kay, B. K. and H. B. Peng (1991). *Xenopus laevis*: practical uses in cell and molecular biology. New York, Academic Press.

Kennedy, M. P., H. Omran, M. W. Leigh, S. Dell, L. Morgan, P. L. Molina, B. V. Robinson, S. L. Minnix, H. Olbrich, T. Severin, P. Ahrens, L. Lange, H. N. Morillas, P. G. Noone, M. A. Zariwala and M. R. Knowles (2007). "Congenital heart disease and other heterotaxic defects in a large cohort of patients with primary ciliary dyskinesia." Circulation **115**(22): 2814-2821.

Kim, H. Y. and L. A. Davidson (2011). "Punctuated actin contractions during convergent extension and their permissive regulation by the non-canonical Wnt-signaling pathway." Journal of Cell Science **124**(4): 635-646.

Kim, H. Y., T. R. Jackson and L. A. Davidson (2016). On the role of mechanics in driving mesenchymal-to-epithelial transitions. Seminars in cell & developmental biology, Elsevier.

Kim, H. Y., T. R. Jackson and L. A. Davidson (2016). "On the role of mechanics in driving mesenchymal-to-epithelial transitions." Semin Cell Dev Biol.

Kim, H. Y., T. R. Jackson, C. Stuckenholz and L. A. Davidson (submitted). "Tissue mechanics drives mesenchymal-to-epithelial transition in embryonic cell aggregates."

Kim, H. Y., T. R. Jackson, C. Stuckenholz and L. A. Davidson (submitted). "Tissue mechanics drives mesenchymal-to-epithelial transition in embryonic cell aggregates.".

Kim, Hye Y., M.-F. Pang, Victor D. Varner, L. Kojima, E. Miller, Derek C. Radisky and Celeste M. Nelson "Localized Smooth Muscle Differentiation Is Essential for Epithelial Bifurcation during Branching Morphogenesis of the Mammalian Lung." Developmental Cell **34**(6): 719-726.

Kim, Y., M. Hazar, D. S. Vijayraghavan, J. Song, T. R. Jackson, S. D. Joshi, W. C. Messner, L. A. Davidson and P. R. LeDuc (2014). "Mechanochemical actuators of embryonic epithelial contractility." Proc Natl Acad Sci U S A **111**(40): 14366-14371.

Kintner, C. (1992). "Regulation of embryonic cell adhesion by the cadherin cytoplasmic domain." Cell **69**(2): 225-236.

Knox, J., K. Moyer, N. Yacoub, C. Soldaat, M. Komosa, K. Vassilieva, R. Wilk, J. Hu, L. d. L. V. Paz and Q. Syed (2011). "Syndecan contributes to heart cell specification and lumen formation during Drosophila cardiogenesis." Developmental biology **356**(2): 279-290.

Kolker, S. J., U. Tajchman and D. L. Weeks (2000). "Confocal imaging of early heart development in Xenopus laevis." Developmental biology **218**: 64-73.

Koser, D. E., A. J. Thompson, S. K. Foster, A. Dwivedy, E. K. Pillai, G. K. Sheridan, H. Svoboda, M. Viana, L. D. Costa, J. Guck, C. E. Holt and K. Franze (2016). "Mechanosensing is critical for axon growth in the developing brain." Nat Neurosci **19**(12): 1592-1598.

Kovacic, J. C., N. Mercader, M. Torres, M. Boehm and V. Fuster (2012). "Epithelial-to-mesenchymal and endothelial-to-mesenchymal transition: from cardiovascular development to disease." Circulation **125**(14): 1795-1808.

Kowalski, P. J., M. A. Rubin and C. G. Kleer (2003). "E-cadherin expression in primary carcinomas of the breast and its distant metastases." Breast Cancer Res **5**(6): R217-R222.

Krens, S. F., S. Mollmert and C. P. Heisenberg (2011). "Enveloping cell-layer differentiation at the surface of zebrafish germ-layer tissue explants." Proc Natl Acad Sci U S A **108**(3): E9-10; author reply E11.

Krieg, M., Y. Arboleda-Estudillo, P. H. Puech, J. Kafer, F. Graner, D. J. Muller and C. P. Heisenberg (2008). "Tensile forces govern germ-layer organization in zebrafish." Nat Cell Biol **10**(4): 429-436.

Kshitiz, M. E. Hubbi, E. H. Ahn, J. Downey, J. Afzal, D. H. Kim, S. Rey, C. Chang, A. Kundu, G. L. Semenza, R. M. Abraham and A. Levchenko (2012). "Matrix rigidity controls endothelial differentiation and morphogenesis of cardiac precursors." Sci Signal **5**(227): ra41.

Kumar, A., M. Ouyang, K. Van den Dries, E. J. McGhee, K. Tanaka, M. D. Anderson, A. Groisman, B. T. Goult, K. I. Anderson and M. A. Schwartz (2016). "Talin tension sensor reveals novel features of focal adhesion force transmission and mechanosensitivity." The Journal of cell biology **213**(3): 371-383.

Kupperman, E., S. An, N. Osborne, S. Waldron and D. Y. Stainier (2000). "A sphingosine-1-phosphate receptor regulates cell migration during vertebrate heart development." Nature **406**(6792): 192-195.

Kuroda, H., L. Fuentealba, A. Ikeda, B. Reversade and E. De Robertis (2005). "Default neural induction: neuralization of dissociated Xenopus cells is mediated by Ras/MAPK activation." Genes & development **19**(9): 1022-1027.

Kurth, T., I. V. Fesenko, S. Schneider, F. E. Munchberg, T. O. Joos, T. P. Spieker and P. Hausen (1999). "Immunocytochemical studies of the interactions of cadherins and catenins in the early Xenopus embryo." Dev Dyn **215**(2): 155-169.

Kwan, K. M. and M. W. Kirschner (2005). "A microtubule-binding Rho-GEF controls cell morphology during convergent extension of Xenopus laevis." Development **132**(20): 4599-4610.

Kwon, J., N. H. Kim and I. Choi (2016). "ROCK activity regulates functional tight junction assembly during blastocyst formation in porcine parthenogenetic embryos." PeerJ **4**: e1914.

Lacalle, R. A., E. Mira, C. Gomez-Mouton, S. Jimenez-Baranda, A. C. Martinez and S. Manes (2002). "Specific SHP-2 partitioning in raft domains triggers integrin-mediated signaling via Rho activation." J Cell Biol **157**(2): 277-289.

Langdon, Y., P. Tandon, E. Paden, J. Duddy, J. M. Taylor and F. L. Conlon (2012). "SHP-2 acts via ROCK to regulate the cardiac actin cytoskeleton." Development **139**(5): 948-957.

Larkin, K. and M. V. Danilchik (1999). "Ventral cell rearrangements contribute to anterior-posterior axis lengthening between neurula and tailbud stages in Xenopus laevis." Dev Biol **216**(2): 550-560.

Latimer, A. and J. R. Jessen (2010). "Extracellular matrix assembly and organization during zebrafish gastrulation." Matrix Biology **29**(2): 89-96.

le Duc, Q., Q. Shi, I. Blonk, A. Sonnenberg, N. Wang, D. Leckband and J. de Rooij (2010). "Vinculin potentiates E-cadherin mechanosensing and is recruited to actin-anchored sites within adherens junctions in a myosin II-dependent manner." J Cell Biol **189**(7): 1107-1115.

Leach, J. B., X. Q. Brown, J. G. Jacot, P. A. Dimilla and J. Y. Wong (2007). "Neurite outgrowth and branching of PC12 cells on very soft substrates sharply decreases below a threshold of substrate rigidity." J Neural Eng **4**(2): 26-34.

Lechuga, S., S. Baranwal and A. I. Ivanov (2015). "Actin-interacting protein 1 controls assembly and permeability of intestinal epithelial apical junctions." Am J Physiol Gastrointest Liver Physiol **308**(9): G745-756.

Lee, Y. H. and J. P. Saint-Jeannet (2011). "Cardiac neural crest is dispensable for outflow tract septation in Xenopus." Development **138**(10): 2025-2034.

Leerberg, J. M., G. A. Gomez, S. Verma, E. J. Moussa, S. K. Wu, R. Priya, B. D. Hoffman, C. Grashoff, M. A. Schwartz and A. S. Yap (2014). "Tension-sensitive actin assembly supports contractility at the epithelial zonula adherens." Current Biology **24**(15): 1689-1699.

Leung, C., Y. Liu, X. Lu, M. Kim, T. A. Drysdale and Q. Feng (2015). "Rac1 Signaling Is Required for Anterior Second Heart Field Cellular Organization and Cardiac Outflow Tract Development." J Am Heart Assoc **5**(1).

Levental, K. R., H. Yu, L. Kass, J. N. Lakins, M. Egeblad, J. T. Erler, S. F. T. Fong, K. Csiszar, A. Giaccia, W. Weninger, M. Yamauchi, D. L. Gasser and V. M. Weaver (2009). "Matrix Crosslinking Forces Tumor Progression by Enhancing Integrin Signaling." Cell **139**(5): 891-906.

Li, R., J. Liang, S. Ni, T. Zhou, X. Qing, H. Li, W. He, J. Chen, F. Li, Q. Zhuang, B. Qin, J. Xu, W. Li, J. Yang, Y. Gan, D. Qin, S. Feng, H. Song, D. Yang, B. Zhang, L. Zeng, L. Lai, M. A. Esteban and D. Pei (2010). "A mesenchymal-to-epithelial transition initiates and is required for the nuclear reprogramming of mouse fibroblasts." Cell Stem Cell **7**(1): 51-63.

Li, S., D. Zhou, M. M. Lu and E. E. Morrisey (2004). "Advanced cardiac morphogenesis does not require heart tube fusion." Science **305**(5690): 1619-1622.

Lin, Y. F., I. Swinburne and D. Yelon (2012). "Multiple influences of blood flow on cardiomyocyte hypertrophy in the embryonic zebrafish heart." Dev Biol **362**(2): 242-253.

Linask, K. K. (1992). "N-cadherin localization in early heart development and polar expression of Na+,K(+)-ATPase, and integrin during pericardial coelom formation and epithelialization of the differentiating myocardium." Dev Biol **151**(1): 213-224.

Linask, K. K. (2003). "Regulation of heart morphology: current molecular and cellular perspectives on the coordinated emergence of cardiac form and function." Birth Defects Res C Embryo Today **69**(1): 14-24.

Linask, K. K., K. A. Knudsen and Y. H. Gui (1997). "N-cadherin-catenin interaction: necessary component of cardiac cell compartmentalization during early vertebrate heart development." Dev Biol **185**(2): 148-164.

Linask, K. K. and J. W. Lash (1993). "Early heart development: dynamics of endocardial cell sorting suggests a common origin with cardiomyocytes." Dev Dyn **196**(1): 62-69.

Linask, K. K., S. Manisastry and M. Han (2005). "Cross talk between cell-cell and cell-matrix adhesion signaling pathways during heart organogenesis: implications for cardiac birth defects." Microscopy and microanalysis : the official journal of Microscopy Society of America, Microbeam Analysis Society, Microscopical Society of Canada **11**: 200-208.

Litman, P., M. R. Amieva and H. Furthmayr (2000). "Imaging of dynamic changes of the actin cytoskeleton in microextensions of live NIH3T3 cells with a GFP fusion of the F-actin binding domain of moesin." BMC Cell Biol **1**: 1.

Liu, Y., N. Pathak, A. Kramer-Zucker and I. A. Drummond (2007). "Notch signaling controls the differentiation of transporting epithelia and multiciliated cells in the zebrafish pronephros." Development **134**(6): 1111-1122.

Lough, J. and Y. Sugi (2000). "Endoderm and heart development." Dev Dyn **217**(4): 327-342.

Lu, M., M. K. Jolly, H. Levine, J. N. Onuchic and E. Ben-Jacob (2013). "MicroRNA-based regulation of epithelial–hybrid–mesenchymal fate determination." Proceedings of the National Academy of Sciences **110**(45): 18144-18149.

Lu, P., K. Takai, V. M. Weaver and Z. Werb (2011). "Extracellular matrix degradation and remodeling in development and disease." Cold Spring Harb Perspect Biol **3**(12).

Lyons, G. E. (1996). "Vertebrate heart development." Curr Opin Genet Dev **6**(4): 454-460.

Maitre, J. L., H. Berthoumieux, S. F. Krens, G. Salbreux, F. Julicher, E. Paluch and C. P. Heisenberg (2012). "Adhesion functions in cell sorting by mechanically coupling the cortices of adhering cells." Science **338**(6104): 253-256.

Majkut, S., P. C. Dingal and D. E. Discher (2014). "Stress sensitivity and mechanotransduction during heart development." Curr Biol **24**(10): R495-501.

Mammoto, T. and D. E. Ingber (2010). "Mechanical control of tissue and organ development." Development (Cambridge, England) **137**: 1407-1420.

Mammoto, T., A. Mammoto, Y.-s. Torisawa, T. Tat, A. Gibbs, R. Derda, R. Mannix, M. de Bruijn, C. W. Yung and D. Huh (2011). "Mechanochemical control of mesenchymal condensation and embryonic tooth organ formation." Developmental cell **21**(4): 758-769.

Manning, M. L., R. A. Foty, M. S. Steinberg and E. M. Schoetz (2010). "Coaction of intercellular adhesion and cortical tension specifies tissue surface tension." Proc Natl Acad Sci U S A **107**(28): 12517-12522.

Markwald, R., C. Eisenberg, L. Eisenberg, T. Trusk and Y. Sugi (1996). "Epithelial-mesenchymal transformations in early avian heart development." Acta Anat (Basel) **156**(3): 173-186.

Marlow, F., J. Topczewski, D. Sepich and L. Solnica-Krezel (2002). "Zebrafish Rho kinase 2 acts downstream of Wnt11 to mediate cell polarity and effective convergence and extension movements." Curr Biol **12**(11): 876-884.

Marsden, M. and D. W. DeSimone (2001). "Regulation of cell polarity, radial intercalation and epiboly in *Xenopus*: novel roles for integrin and fibronectin." Development **128**: 3635-3647.

Maruthamuthu, V., B. Sabass, U. S. Schwarz and M. L. Gardel (2011). "Cell-ECM traction force modulates endogenous tension at cell-cell contacts." Proc Natl Acad Sci U S A **108**(12): 4708-4713.

McBeath, R., D. M. Pirone, C. M. Nelson, K. Bhadriraju and C. S. Chen (2004). "Cell Shape, Cytoskeletal Tension, and RhoA Regulate Stem Cell Lineage Commitment." Developmental Cell **6**(4): 483-495.

Medioni, C., M. Astier, M. Zmojdzian, K. Jagla and M. Semeriva (2008). "Genetic control of cell morphogenesis during Drosophila melanogaster cardiac tube formation." J Cell Biol **182**(2): 249-261.

Medioni, C., M. Astier, M. Zmojdzian, K. Jagla and M. Sémériva (2008). "Genetic control of cell morphogenesis during Drosophila melanogaster cardiac tube formation." The Journal of cell biology **182**(2): 249-261.

Meyer, T. N., C. Schwesinger, R. V. Sampogna, D. A. Vaughn, R. O. Stuart, D. L. Steer, K. T. Bush and S. K. Nigam (2006). "Rho kinase acts at separate steps in ureteric bud and metanephric mesenchyme morphogenesis during kidney development." Differentiation **74**(9-10): 638-647.

Miller, C. J. and L. A. Davidson (2013). "The interplay between cell signalling and mechanics in developmental processes." Nat Rev Genet **14**(10): 733-744.

Miller, C. J. and L. A. Davidson (2013). "The interplay between cell signalling and mechanics in developmental processes." Nature Reviews Genetics **14**(10): 733-744.

Mitchell, B., R. Jacobs, J. Li, S. Chien and C. Kintner (2007). "A positive feedback mechanism governs the polarity and motion of motile cilia." Nature **447**(7140): 97-101.

Mitchell, B., J. L. Stubbs, F. Huisman, P. Taborek, C. Yu and C. Kintner (2009). "The PCP pathway instructs the planar orientation of ciliated cells in the Xenopus larval skin." Current Biology **19**(11): 924-929.

Mohun, T. J., L. M. Leong, W. J. Weninger and D. B. Sparrow (2000). "The morphology of heart development in Xenopus laevis." Dev Biol **218**(1): 74-88.

Moore, S. W. (1994). "A fiber optic system for measuring dynamic mechanical properties of embryonic tissues." IEEE Transaction on Biomedical Engineering **41**(1): 45-50.

Moore, S. W., R. E. Keller and M. A. Koehl (1995). "The dorsal involuting marginal zone stiffens anisotropically during its convergent extension in the gastrula of Xenopus laevis." Development **121**(10): 3131-3140.

Moore, S. W., R. E. Keller and M. A. R. Koehl (1995). "The dorsal involuting marginal zone stiffens anisotropically during its convergent extension in the gastrula of *Xenopus leavis*." Development **121**(10): 3130-3140.

Morikawa, Y., M. Zhang, T. Heallen, J. Leach, G. Tao, Y. Xiao, Y. Bai, W. Li, J. T. Willerson and J. F. Martin (2015). "Actin cytoskeletal remodeling with protrusion formation is essential for heart regeneration in Hippo-deficient mice." Sci Signal **8**(375): ra41.

Morris, S. A. (2011). "Cell fate in the early mouse embryo: sorting out the influence of developmental history on lineage choice." Reproductive biomedicine online **22**(6): 521-524.

Mulligan, T., H. Blaser, E. Raz and S. A. Farber (2010). "Prenylation-deficient G protein gamma subunits disrupt GPCR signaling in the zebrafish." Cell Signal **22**: 221-233.

Munro, E., J. Nance and J. R. Priess (2004). "Cortical flows powered by asymmetrical contraction transport PAR proteins to establish and maintain anterior-posterior polarity in the early C. elegans embryo." Developmental cell **7**(3): 413-424.

Muranyi, A., D. Derkach, F. Erdodi, A. Kiss, M. Ito and D. J. Hartshorne (2005). "Phosphorylation of Thr695 and Thr850 on the myosin phosphatase target subunit: inhibitory effects and occurrence in A7r5 cells." FEBS Lett **579**(29): 6611-6615.

Nagelkerke, A., J. Bussink, A. E. Rowan and P. N. Span (2015). "The mechanical microenvironment in cancer: How physics affects tumours." Semin Cancer Biol **35**: 62-70.

Nakaya, Y., S. Kuroda, Y. T. Katagiri, K. Kaibuchi and Y. Takahashi (2004). "Mesenchymal-epithelial transition during somitic segmentation is regulated by differential roles of Cdc42 and Rac1." Developmental cell **7**(3): 425-438.

Nance, J. (2014). "Getting to know your neighbor: cell polarization in early embryos." J Cell Biol **206**(7): 823-832.

Nance, J., E. M. Munro and J. R. Priess (2003). "C. elegans PAR-3 and PAR-6 are required for apicobasal asymmetries associated with cell adhesion and gastrulation." Development **130**(22): 5339-5350.

Nascone, N. and M. Mercola (1995). "An inductive role for the endoderm in Xenopus cardiogenesis." Development **121**(2): 515-523.

Nelson, C. M. and J. P. Gleghorn (2012). "Sculpting organs: mechanical regulation of tissue development." Annu Rev Biomed Eng **14**: 129-154.

Nembhard, W. N., T. Wang, M. L. Loscalzo and J. L. Salemi (2010). "Variation in the prevalence of congenital heart defects by maternal race/ethnicity and infant sex." J Pediatr **156**(2): 259-264.

Nieto, M. A. (2013). "Epithelial plasticity: a common theme in embryonic and cancer cells." Science **342**(6159): 1234850.

Nieto, M. A., M. G. Sargent, D. G. Wilkinson and J. Cooke (1994). "Control of cell behavior during vertebrate development by Slug, a zinc finger gene." Science **264**(5160): 835-839.

Nieuwkoop, P. D. and J. Faber (1967). Normal tables of *Xenopus laevis* (Daudin). Amsterdam, Elsevier North-Holland Biomedical Press.

Noel, E. S., M. Verhoeven, A. K. Lagendijk, F. Tessadori, K. Smith, S. Choorapoikayil, J. den Hertog and J. Bakkers (2013). "A Nodal-independent and tissue-intrinsic mechanism controls heart-looping chirality." Nat Commun **4**: 2754.

Noonan, J. A. (1994). "Noonan syndrome. An update and review for the primary pediatrician." Clin Pediatr (Phila) **33**(9): 548-555.

Ocaña, Oscar H., R. Córcoles, Á. Fabra, G. Moreno-Bueno, H. Acloque, S. Vega, A. Barrallo-Gimeno, A. Cano and M. A. Nieto "Metastatic Colonization Requires the Repression of the Epithelial-Mesenchymal Transition Inducer Prrx1." Cancer Cell **22**(6): 709-724.

Oh, E. C. and N. Katsanis (2012). "Cilia in vertebrate development and disease." Development **139**(3): 443-448.

Ohta, S., K. Suzuki, K. Tachibana, H. Tanaka and G. Yamada (2007). "Cessation of gastrulation is mediated by suppression of epithelial-mesenchymal transition at the ventral ectodermal ridge." Development **134**(24): 4315-4324.

Oster, G. F., J. D. Murray and A. K. Harris (1983). "Mechanical aspects of mesenchymal morphogenesis." Journal of Embryology and Experimental Morphology **78**: 83-125.

Palmeirim, I., D. Henrique, D. Ish-Horowicz and O. Pourquie (1997). "Avian hairy gene expression identifies a molecular clock linked to vertebrate segmentation and somitogenesis." Cell **91**(5): 639-648.

Pandur, P., I. O. Sirbu, S. J. Kuhl, M. Philipp and M. Kuhl (2013). "Islet1-expressing cardiac progenitor cells: a comparison across species." Dev Genes Evol **223**(1-2): 117-129.

Paszek, M. J. and V. M. Weaver (2004). "The tension mounts: mechanics meets morphogenesis and malignancy." J Mammary Gland Biol Neoplasia **9**(4): 325-342.

Pattabiraman, D. R., B. Bierie, K. I. Kober, P. Thiru, J. A. Krall, C. Zill, F. Reinhardt, W. L. Tam and R. A. Weinberg (2016). "Activation of PKA leads to mesenchymal-to-epithelial transition and loss of tumor-initiating ability." Science **351**(6277): aad3680.

Pérez-Pomares, J. M. and R. Muñoz-Chápuli (2002). "Epithelial–mesenchymal transitions: a mesodermal cell strategy for evolutive innovation in metazoans." The Anatomical Record **268**(3): 343-351.

Pierpont, M. E., C. T. Basson, D. W. Benson, Jr., B. D. Gelb, T. M. Giglia, E. Goldmuntz, G. McGee, C. A. Sable, D. Srivastava, C. L. Webb and C. o. C. D. i. t. Y. American Heart Association Congenital Cardiac Defects Committee (2007). "Genetic basis for congenital heart defects: current knowledge: a scientific statement from the American Heart Association Congenital Cardiac Defects Committee, Council on Cardiovascular Disease in the Young: endorsed by the American Academy of Pediatrics." Circulation **115**(23): 3015-3038.

Planas-Silva, M. D. and P. K. Waltz (2007). "Estrogen promotes reversible epithelial-to-mesenchymal-like transition and collective motility in MCF-7 breast cancer cells." The Journal of steroid biochemistry and molecular biology **104**(1): 11-21.

Plusa, B., S. Frankenberg, A. Chalmers, A. K. Hadjantonakis, C. A. Moore, N. Papalopulu, V. E. Papaioannou, D. M. Glover and M. Zernicka-Goetz (2005). "Downregulation of Par3 and aPKC function directs cells towards the ICM in the preimplantation mouse embryo." J Cell Sci **118**(Pt 3): 505-515.

Preibisch, S., S. Saalfeld and P. Tomancak (2009). "Globally optimal stitching of tiled 3D microscopic image acquisitions." Bioinformatics **25**(11): 1463-1465.

Puklin-Faucher, E., M. Gao, K. Schulten and V. Vogel (2006). "How the headpiece hinge angle is opened: New insights into the dynamics of integrin activation." J Cell Biol **175**(2): 349-360.

Puri, P. and W. H. Walker (2013). "The tyrosine phosphatase SHP2 regulates Sertoli cell junction complexes." Biol Reprod **88**(3): 59.

Quail, D. F. and J. A. Joyce (2013). "Microenvironmental regulation of tumor progression and metastasis." Nat Med **19**(11): 1423-1437.

Raab, M., J. Swift, P. C. Dingal, P. Shah, J. W. Shin and D. E. Discher (2012). "Crawling from soft to stiff matrix polarizes the cytoskeleton and phosphoregulates myosin-II heavy chain." J Cell Biol **199**(4): 669-683.

Radice, G. L., H. Rayburn, H. Matsunami, K. A. Knudsen, M. Takeichi and R. O. Hynes (1997). "Developmental defects in mouse embryos lacking N-cadherin." Dev Biol **181**(1): 64-78.

Ramos, J. W. and D. W. DeSimone (1996). "*Xenopus* embryonic cell adhesion to fibronectin: position-specific activation of RGD / Synergy site-dependent migratory behavior at gastrulation." J Cell Biol **134**(1): 1-14.

Ramsbottom, S. A., V. Sharma, H. J. Rhee, L. Eley, H. M. Phillips, H. F. Rigby, C. Dean, B. Chaudhry and D. J. Henderson (2014). "Vangl2-Regulated Polarisation of Second Heart Field-Derived Cells Is Required for Outflow Tract Lengthening during Cardiac Development." PLoS genetics **10**: e1004871.

Ransom, D. G., M. D. Hens and D. W. DeSimone (1993). "Integrin expression in early amphibian embryos: cDNA cloning and characterization of Xenopus beta 1, beta 2, beta 3, and beta 6 subunits." Dev Biol **160**(1): 265-275.

Remond, M. C., J. A. Fee, E. L. Elson and L. A. Taber (2006). "Myosin-based contraction is not necessary for cardiac c-looping in the chick embryo." Anat Embryol (Berl) **211**(5): 443-454.

Reuter, R., B. Grunewald and M. Leptin (1993). "A role for the mesoderm in endodermal migration and morphogenesis in Drosophila." Development **119**(4): 1135-1145.

Roberts, S. J., D. S. Leaf, H. P. Moore and J. C. Gerhart (1992). "The establishment of polarized membrane traffic in Xenopus laevis embryos." J Cell Biol **118**(6): 1359-1369.

Rohr, S., N. Bit-Avragim and S. Abdelilah-Seyfried (2006). "Heart and soul/PRKCi and nagie oko/Mpp5 regulate myocardial coherence and remodeling during cardiac morphogenesis." Development **133**(1): 107-115.

Rozario, T., B. Dzamba, G. F. Weber, L. A. Davidson and D. W. DeSimone (2009). "The physical state of fibronectin matrix differentially regulates morphogenetic movements in vivo." Dev Biol **327**(2): 386-398.

Rugendorff, A., A. Younossi-Hartenstein and V. Hartenstein (1994). "Embryonic origin and differentiation of the Drosophila heart." Roux's archives of developmental biology **203**(5): 266-280.

Sabass, B., M. L. Gardel, C. M. Waterman and U. S. Schwarz (2008). "High resolution traction force microscopy based on experimental and computational advances." Biophysical journal **94**: 207-220.

Saiz, N., J. B. Grabarek, N. Sabherwal, N. Papalopulu and B. Plusa (2013). "Atypical protein kinase C couples cell sorting with primitive endoderm maturation in the mouse blastocyst." Development **140**(21): 4311-4322.

Sakurai, K., I. Talukdar, Veena S. Patil, J. Dang, Z. Li, K.-Y. Chang, C.-C. Lu, V. Delorme-Walker, C. DerMardirossian, K. Anderson, D. Hanein, C.-S. Yang, D. Wu, Y. Liu and

Tariq M. Rana (2014). "Kinome-wide Functional Analysis Highlights the Role of Cytoskeletal Remodeling in Somatic Cell Reprogramming." Cell stem cell **14**(4): 523-534.

Samavarchi-Tehrani, P., A. Golipour, L. David, H. K. Sung, T. A. Beyer, A. Datti, K. Woltjen, A. Nagy and J. L. Wrana (2010). "Functional genomics reveals a BMP-driven mesenchymal-to-epithelial transition in the initiation of somatic cell reprogramming." Cell Stem Cell **7**(1): 64-77.

Samuel, M. S., J. I. Lopez, E. J. McGhee, D. R. Croft, D. Strachan, P. Timpson, J. Munro, E. Schroder, J. Zhou, V. G. Brunton, N. Barker, H. Clevers, O. J. Sansom, K. I. Anderson, V. M. Weaver and M. F. Olson (2011). "Actomyosin-mediated cellular tension drives increased tissue stiffness and beta-catenin activation to induce epidermal hyperplasia and tumor growth." Cancer Cell **19**(6): 776-791.

Samuel, Michael S., Jose I. Lopez, Ewan J. McGhee, Daniel R. Croft, D. Strachan, P. Timpson, J. Munro, E. Schröder, J. Zhou, Valerie G. Brunton, N. Barker, H. Clevers, Owen J. Sansom, Kurt I. Anderson, Valerie M. Weaver and Michael F. Olson (2011). "Actomyosin-Mediated Cellular Tension Drives Increased Tissue Stiffness and β-Catenin Activation to Induce Epidermal Hyperplasia and Tumor Growth." Cancer Cell **19**(6): 776-791.

Santhanakrishnan, A. and L. A. Miller (2011). "Fluid dynamics of heart development." Cell Biochem Biophys **61**(1): 1-22.

Santos, O. F. and S. K. Nigam (1993). "HGF-induced tubulogenesis and branching of epithelial cells is modulated by extracellular matrix and TGF-β." Developmental biology **160**(2): 293-302.

Sater, A. K., R. A. Steinhardt and R. Keller (1993). "Induction of neuronal differentiation by planar signals in Xenopus embryos." Dev Dyn **197**(4): 268-280.

Sato, M., M. J. Levesque and R. M. Nerem (1987). "An application of the micropipette technique to the measurement of the mechanical properties of cultured bovine aortic endothelial cells." Journal of Biomechanical Engineering **109(1)**: 27-34.

Schaeffer, D., J. A. Somarelli, G. Hanna, G. M. Palmer and M. A. Garcia-Blanco (2014). "Cellular migration and invasion uncoupled: increased migration is not an inexorable consequence of epithelial-to-mesenchymal transition." Mol Cell Biol **34**(18): 3486-3499.

Schedin, P. and P. J. Keely (2011). "Mammary gland ECM remodeling, stiffness, and mechanosignaling in normal development and tumor progression." Cold Spring Harb Perspect Biol **3**(1): a003228.

Schneider, C. A., W. S. Rasband and K. W. Eliceiri (2012). "NIH Image to ImageJ: 25 years of image analysis." Nat methods **9**(7): 671-675.

Schneider, V. A. and M. Mercola (1999). "Spatially distinct head and heart inducers within the Xenopus organizer region." Curr Biol **9**(15): 800-809.

Schneider, V. A. and M. Mercola (2001). "Wnt antagonism initiates cardiogenesis in Xenopus laevis." Genes Dev **15**(3): 304-315.

Schonegg, S. and A. A. Hyman (2006). "CDC-42 and RHO-1 coordinate acto-myosin contractility and PAR protein localization during polarity establishment in C. elegans embryos." Development **133**(18): 3507-3516.

Schrode, N., P. Xenopoulos, A. Piliszek, S. Frankenberg, B. Plusa and A. K. Hadjantonakis (2013). "Anatomy of a blastocyst: cell behaviors driving cell fate choice and morphogenesis in the early mouse embryo." Genesis **51**(4): 219-233.

Schwartz, M. A. and D. W. DeSimone (2008). "Cell adhesion receptors in mechanotransduction." Curr Opin Cell Biol **20**(5): 551-556.

Sedzinski, J., E. Hannezo, F. Tu, M. Biro and J. B. Wallingford (2016). "Emergence of an Apical Epithelial Cell Surface In Vivo." Developmental cell **36**(1): 24-35.

Sharland, M., M. Burch, W. M. McKenna and M. A. Paton (1992). "A clinical study of Noonan syndrome." Arch Dis Child **67**(2): 178-183.

Shawky, J. H. and L. A. Davidson (2015). "Tissue mechanics and adhesion during embryo development." Dev Biol **401**(1): 152-164.

Shi, Y., V. D. Varner and L. A. Taber (2015). "Why is cytoskeletal contraction required for cardiac fusion before but not after looping begins?" Phys Biol **12**(1): 016012.

Shibue, T. and R. A. Weinberg (2011). "Metastatic colonization: Settlement, adaptation and propagation of tumor cells in a foreign tissue environment." Seminars in Cancer Biology **21**(2): 99-106.

Shook, D. and R. Keller (2003). "Mechanisms, mechanics and function of epithelial-mesenchymal transitions in early development." Mech Dev **120**(11): 1351-1383.

Shook, D. and R. Keller (2003). "Mechanisms, mechanics and function of epithelial–mesenchymal transitions in early development." Mechanisms of Development **120**(11): 1351-1383.

Siliciano, J. D. and D. A. Goodenough (1988). "Localization of the tight junction protein, ZO-1, is modulated by extracellular calcium and cell-cell contact in Madin-Darby canine kidney epithelial cells." J Cell Biol **107**(6 Pt 1): 2389-2399.

Šindelka, R., Z. Ferjentsik and J. Jonák (2006). "Developmental expression profiles of Xenopus laevis reference genes." Developmental Dynamics **235**(3): 754-758.

Sive, H. L., R. M. Grainger and R. M. Harland (2000). Early development of Xenopus laevis: a laboratory manual. Cold Spring Harbor, New York, Cold Spring Harbor Laboratory Press.

Song, W., E. Dyer, D. J. Stuckey, O. Copeland, M. C. Leung, C. Bayliss, A. Messer, R. Wilkinson, J. L. Tremoleda, M. D. Schneider, S. E. Harding, C. S. Redwood, K. Clarke, K. Nowak, L. Monserrat, D. Wells and S. B. Marston (2011). "Molecular mechanism of the E99K mutation in cardiac actin (ACTC Gene) that causes apical hypertrophy in man and mouse." J Biol Chem **286**(31): 27582-27593.

Sosa, M. S., P. Bragado and J. A. Aguirre-Ghiso (2014). "Mechanisms of disseminated cancer cell dormancy: an awakening field." Nat Rev Cancer **14**(9): 611-622.

Soufan, A. T., G. van den Berg, J. M. Ruijter, P. A. de Boer, M. J. van den Hoff and A. F. Moorman (2006). "Regionalized sequence of myocardial cell growth and proliferation characterizes early chamber formation." Circ Res **99**(5): 545-552.

Stark, K., S. Vainio, G. Vassileva and A. P. McMahon (1994). "Epithelial transformation of metanephric mesenchyme in the developing kidney regulated by Wnt-4." Nature **372**(6507): 679-683.

Steinberg, M. S. and S. F. Gilbert (2004). "Townes and Holtfreter (1955): directed movements and selective adhesion of embryonic amphibian cells." J Exp Zoolog A Comp Exp Biol **301**(9): 701-706.

Straight, A. F., A. Cheung, J. Limouze, I. Chen, N. J. Westwood, J. R. Sellers and T. J. Mitchison (2003). "Dissecting temporal and spatial control of cytokinesis with a myosin II Inhibitor." Science **299**(5613): 1743-1747.

Stubbs, J. L., L. Davidson, R. Keller and C. Kintner (2006). "Radial intercalation of ciliated cells during Xenopus skin development." Development **133**(13): 2507-2515.

Sugi, Y. and R. R. Markwald (1996). "Formation and early morphogenesis of endocardial endothelial precursor cells and the role of endoderm." Dev Biol **175**(1): 66-83.

Sutton, M., T. Plappert, A. Spiegel, J. Raichlen, P. Douglas, N. Reichek and L. Edmunds (1987). "Early postoperative changes in left ventricular chamber size, architecture, and function in aortic stenosis and aortic regurgitation and their relation to intraoperative changes in afterload: a prospective two-dimensional echocardiographic study." Circulation **76**(1): 77-89.

Szabo, A. and R. Mayor (2016). "Modelling collective cell migration of neural crest." Curr Opin Cell Biol **42**: 22-28.

Szabo, A., M. Melchionda, G. Nastasi, M. L. Woods, S. Campo, R. Perris and R. Mayor (2016). "In vivo confinement promotes collective migration of neural crest cells." J Cell Biol **213**(5): 543-555.

Taber, L. A. (2001). "Biomechanics of cardiovascular development." Annu Rev Biomed Eng **3**: 1-25.

Taber, L. A. (2014). "Morphomechanics: transforming tubes into organs." Curr Opin Genet Dev **27**: 7-13.

Taber, L. A., J. Zhang and R. Perucchio (2007). "Computational model for the transition from peristaltic to pulsatile flow in the embryonic heart tube." J Biomech Eng **129**(3): 441-449.

Takahashi, K. and S. Yamanaka (2006). "Induction of Pluripotent Stem Cells from Mouse Embryonic and Adult Fibroblast Cultures by Defined Factors." Cell **126**(4): 663-676.

Tambe, D. T., U. Croutelle, X. Trepat, C. Y. Park, J. H. Kim, E. Millet, J. P. Butler and J. J. Fredberg (2013). "Monolayer stress microscopy: limitations, artifacts, and accuracy of recovered intercellular stresses." PLoS One **8**(2): e55172.

Tartaglia, M., E. L. Mehler, R. Goldberg, G. Zampino, H. G. Brunner, H. Kremer, I. van der Burgt, A. H. Crosby, A. Ion, S. Jeffery, K. Kalidas, M. A. Patton, R. S. Kucherlapati and B. D. Gelb (2001). "Mutations in PTPN11, encoding the protein tyrosine phosphatase SHP-2, cause Noonan syndrome." Nat Genet **29**(4): 465-468.

Tepass, U. (1997). "Epithelial differentiation in Drosophila." Bioessays **19**(8): 673-682.

Tepass, U. and V. Hartenstein (1995). "Neurogenic and proneural genes control cell fate specification in the Drosophila endoderm." Development **121**(2): 393-405.

Tepass, U., G. Tanentzapf, R. Ward and R. Fehon (2001). "Epithelial cell polarity and cell junctions in Drosophila." Annu Rev Genet **35**: 747-784.

Thiery, J. P., H. Acloque, R. Y. Huang and M. A. Nieto (2009). "Epithelial-mesenchymal transitions in development and disease." Cell **139**(5): 871-890.

Thiery, J. P., H. Acloque, R. Y. J. Huang and M. A. Nieto (2009). "Epithelial-Mesenchymal Transitions in Development and Disease." Cell **139**(5): 871-890.

Tien, S. C., H. H. Lee, Y. C. Yang, M. H. Lin, Y. J. Chen and Z. F. Chang (2016). "The Shp2-induced epithelial disorganization defect is reversed by HDAC6 inhibition independent of Cdc42." Nat Commun **7**: 10420.

Trepat, X., M. R. Wasserman, T. E. Angelini, E. Millet, D. A. Weitz, J. P. Butler and J. J. Fredberg (2009). "Physical forces during collective cell migration." Nature Physics **5**(6): 426-430.

Trichet, L., J. Le Digabel, R. J. Hawkins, S. R. Vedula, M. Gupta, C. Ribrault, P. Hersen, R. Voituriez and B. Ladoux (2012). "Evidence of a large-scale mechanosensing mechanism for cellular adaptation to substrate stiffness." Proc Natl Acad Sci U S A **109**(18): 6933-6938.

Trinh, L. A. and D. Y. Stainier (2004). "Fibronectin regulates epithelial organization during myocardial migration in zebrafish." Dev Cell **6**(3): 371-382.

Trinh, L. A. and D. Y. R. Stainier (2004). "Fibronectin regulates epithelial organization during myocardial migration in zebrafish." Developmental Cell **6**: 371-382.

Trinh, L. A., D. Yelon and D. Y. Stainier (2005). "Hand2 regulates epithelial formation during myocardial diferentiation." Curr Biol **15**(5): 441-446.

Trinh, L. A., D. Yelon and D. Y. Stainier (2005). "Hand2 regulates epithelial formation during myocardial differentiation." Current biology **15**(5): 441-446.

Truskinovsky, L., G. Vitale and T. Smit (2014). "A mechanical perspective on vertebral segmentation." International Journal of Engineering Science **83**: 124-137.

Tsai, Jeff H., Joana L. Donaher, Danielle A. Murphy, S. Chau and J. Yang (2012). "Spatiotemporal Regulation of Epithelial-Mesenchymal Transition Is Essential for Squamous Cell Carcinoma Metastasis." Cancer Cell **22**(6): 725-736.

Ulrich, F., M. L. Concha, P. J. Heid, E. Voss, S. Witzel, H. Roehl, M. Tada, S. W. Wilson, R. J. Adams, D. R. Soll and C. P. Heisenberg (2003). "Slb/Wnt11 controls hypoblast cell migration and morphogenesis at the onset of zebrafish gastrulation." Development **130**(22): 5375-5384.

Valentine, J. W. (1978). "The evolution of multicellular plants and animals." Scientific American **239**(3): 141-158.

Varner, V. D. and L. A. Taber (2012). "Not just inductive: a crucial mechanical role for the endoderm during heart tube assembly." Development **139**(9): 1680-1690.

Varner, V. D. and L. a. Taber (2012). "Not just inductive: a crucial mechanical role for the endoderm during heart tube assembly." Development (Cambridge, England) **139**: 1680-1690.

Vasilyev, A., Y. Liu, N. Hellman, N. Pathak and I. A. Drummond (2012). "Mechanical stretch and PI3K signaling link cell migration and proliferation to coordinate epithelial tubule morphogenesis in the zebrafish pronephros." PloS one **7**(7): e39992.

Vermot, J., A. S. Forouhar, M. Liebling, D. Wu, D. Plummer, M. Gharib and S. E. Fraser (2009). "Reversing blood flows act through klf2a to ensure normal valvulogenesis in the developing heart." PLoS Biol **7**(11): e1000246.

Vincent, R., E. Bazellieres, C. Perez-Gonzalez, M. Uroz, X. Serra-Picamal and X. Trepat (2015). "Active Tensile Modulus of an Epithelial Monolayer." Phys Rev Lett **115**(24): 248103.

von Dassow, M. and L. A. Davidson (2009). "Natural variation in embryo mechanics: gastrulation in Xenopus laevis is highly robust to variation in tissue stiffness." Dev Dyn **238**(1): 2-18.

von Dassow, M. and L. A. Davidson (2009). "Natural variation in embryo mechanics: gastrulation in Xenopus laevis is highly robust to variation in tissue stiffness." Dev Dyn **238**: 2-18.

von Dassow, M., C. J. Miller and L. A. Davidson (2014). "Biomechanics and the thermotolerance of development." PLoS One **9**(4): e95670.

von Dassow, M., J. A. Strother and L. A. Davidson (2010). "Surprisingly simple mechanical behavior of a complex embryonic tissue." PLoS One **5**(12): e15359.

von Gise, A. and W. T. Pu (2012). "Endocardial and epicardial epithelial to mesenchymal transitions in heart development and disease." Circulation research **110**: 1628-1645.

Waldo, K., S. Miyagawa-Tomita, D. Kumiski and M. L. Kirby (1998). "Cardiac neural crest cells provide new insight into septation of the cardiac outflow tract: aortic sac to ventricular septal closure." Dev Biol **196**(2): 129-144.

Wallingford, J. B. and R. M. Harland (2001). "Xenopus Dishevelled signaling regulates both neural and mesodermal convergent extension: parallel forces elongating the body axis." Development **128**(13): 2581-2592.

Walsh, S. V., A. M. Hopkins, J. Chen, S. Narumiya, C. A. Parkos and A. Nusrat (2001). "Rho kinase regulates tight junction function and is necessary for tight junction assembly in polarized intestinal epithelia." Gastroenterology **121**(3): 566-579.

Wang, H. B., M. Dembo, S. K. Hanks and Y. Wang (2001). "Focal adhesion kinase is involved in mechanosensing during fibroblast migration." Proc Natl Acad Sci U S A **98**(20): 11295-11300.

Wang, S., S.-W. Cha, A. M. Zorn and C. Wylie (2013). "Par6b regulates the dynamics of apicobasal polarity during development of the stratified Xenopus epidermis." PloS one **8**(10): e76854.

Warkman, A. S. and P. A. Krieg (2007). "Xenopus as a model system for vertebrate heart development." Semin Cell Dev Biol **18**(1): 46-53.

Weber, G. F., M. A. Bjerke and D. W. DeSimone (2012). "A mechanoresponsive cadherin-keratin complex directs polarized protrusive behavior and collective cell migration." Dev Cell **22**(1): 104-115.

Wei, L., W. Roberts, L. Wang, M. Yamada, S. Zhang, Z. Zhao, S. A. Rivkees, R. J. Schwartz and K. Imanaka-Yoshida (2001). "Rho kinases play an obligatory role in vertebrate embryonic organogenesis." Development **128**(15): 2953-2962.

White, M. D. and N. Plachta (2015). "How Adhesion Forms the Early Mammalian Embryo." Current topics in developmental biology **112**: 1-17.

Winkel, G. K., J. E. Ferguson, M. Takeichi and R. Nuccitelli (1990). "Activation of protein kinase C triggers premature compaction in the four-cell stage mouse embryo." Dev Biol **138**(1): 1-15.

Witzel, H. R., S. Cheedipudi, R. Gao, D. Y. Stainier and G. D. Dobreva (2017). "Isl2b regulates anterior second heart field development in zebrafish." Sci Rep **7**: 41043.

Wozniak, M. A. and C. S. Chen (2009). "Mechanotransduction in development: a growing role for contractility." Nat Rev Mol Cell Biol **10**(1): 34-43.

Xie, H., D. Ye, D. Sepich and F. Lin (2016). "S1pr2/Gα13 signaling regulates the migration of endocardial precursors by controlling endoderm convergence." Developmental Biology **414**: 228-243.

Yam, P. T., C. A. Wilson, L. Ji, B. Hebert, E. L. Barnhart, N. A. Dye, P. W. Wiseman, G. Danuser and J. A. Theriot (2007). "Actin-myosin network reorganization breaks symmetry at the cell rear to spontaneously initiate polarized cell motility." J Cell Biol **178**(7): 1207-1221.

Yamada, S. and W. J. Nelson (2007). "Localized zones of Rho and Rac activities drive initiation and expansion of epithelial cell-cell adhesion." J Cell Biol **178**(3): 517-527.

Yamada, S. and W. J. Nelson (2007). "Localized zones of Rho and Rac activities drive initiation and expansion of epithelial cell–cell adhesion." The Journal of Cell Biology **178**(3): 517-527.

Yan, J., M. Yao, B. T. Goult and M. P. Sheetz (2015). "Talin Dependent Mechanosensitivity of Cell Focal Adhesions." Cell Mol Bioeng **8**(1): 151-159.

Yang, J. and R. A. Weinberg (2008). "Epithelial-Mesenchymal Transition: At the Crossroads of Development and Tumor Metastasis." Developmental Cell **14**(6): 818-829.

Yarnitzky, T. and T. Volk (1995). "Laminin is required for heart, somatic muscles, and gut development in the Drosophila embryo." Developmental biology **169**(2): 609-618.

Yates, L. L., J. Papakrivopoulou, D. A. Long, P. Goggolidou, J. O. Connolly, A. S. Woolf and C. H. Dean (2010). "The planar cell polarity gene Vangl2 is required for mammalian kidney-branching morphogenesis and glomerular maturation." Human molecular genetics **19**(23): 4663-4676.

Ye, D. and F. Lin (2013). "S1pr2/Galpha13 signaling controls myocardial migration by regulating endoderm convergence." Development **140**(4): 789-799.

Ye, D. and F. Lin (2013). "S1pr2/Gα13 signaling controls myocardial migration by regulating endoderm convergence." Development (Cambridge, England) **140**: 789-799.

Ye, D., H. Xie, B. Hu and F. Lin (2015). "Endoderm convergence controls subduction of the myocardial precursors during heart-tube formation." Development (Cambridge, England) **142**: 2928-2940.

Yelon, D. and D. Y. R. Stainier (2005). "Hand2 Regulates Epithelial Formation during Myocardial Differentiation." Current Biology **15**: 441-446.

Yin, C., M. Kiskowski, P. A. Pouille, E. Farge and L. Solnica-Krezel (2008). "Cooperation of polarized cell intercalations drives convergence and extension of presomitic mesoderm during zebrafish gastrulation." J Cell Biol **180**(1): 221-232.

Yoo, T. S., M. J. Ackerman, W. E. Lorensen, W. Schroeder, V. Chalana, S. Aylward, D. Metaxas and R. Whitaker (2002). "Engineering and algorithm design for an image processing Api: a technical report on ITK--the Insight Toolkit." Stud Health Technol Inform **85**: 586-592.

Zamir, E. A., A. Czirók, C. Cui, C. D. Little and B. J. Rongish (2006). "Mesodermal cell displacements during avian gastrulation are due to both individual cell-autonomous and convective tissue movements." Proceedings of the National Academy of Sciences of the United States of America **103**: 19806-19811.

Zamir, E. A. and L. A. Taber (2004). "Material properties and residual stress in the stage 12 chick heart during cardiac looping." J Biomech Eng **126**(6): 823-830.

Zeisberg, M., A. A. Shah and R. Kalluri (2005). "Bone morphogenic protein-7 induces mesenchymal to epithelial transition in adult renal fibroblasts and facilitates regeneration of injured kidney." Journal of Biological Chemistry **280**(9): 8094-8100.

Zhang, J., M. Betson, J. Erasmus, K. Zeikos, M. Bailly, L. P. Cramer and V. M. M. Braga (2005). "Actin at cell-cell junctions is composed of two dynamic and functional populations." Journal of Cell Science **118**(23): 5549-5562.

Zhang, J., X.-J. Tian, H. Zhang, Y. Teng, R. Li, F. Bai, S. Elankumaran and J. Xing (2014). "TGF-β-induced epithelial-to-mesenchymal transition proceeds through stepwise activation of multiple feedback loops." Sci Signal **7**(345): ra91.

Zhang, K., H. Zhao, Z. Ji, C. Zhang, P. Zhou, L. Wang, Q. Chen, J. Wang, P. Zhang, Z. Chen, H. H. Zhu and W. Q. Gao (2016). "Shp2 promotes metastasis of prostate cancer by attenuating the PAR3/PAR6/aPKC polarity protein complex and enhancing epithelial-to-mesenchymal transition." Oncogene **35**(10): 1271-1282.

Zhang, P., S. Wang, S. Wang, J. Qiao, L. Zhang, Z. Zhang and Z. Chen (2016). "Dual function of partitioning-defective 3 in the regulation of YAP phosphorylation and activation." Cell Discov **2**: 16021.

Zhang, W., R. J. Chan, H. Chen, Z. Yang, Y. He, X. Zhang, Y. Luo, F. Yin, A. Moh, L. C. Miller, R. M. Payne, Z. Y. Zhang, X. Y. Fu and W. Shou (2009). "Negative regulation of

Stat3 by activating PTPN11 mutants contributes to the pathogenesis of Noonan syndrome and juvenile myelomonocytic leukemia." J Biol Chem **284**(33): 22353-22363.

Zhou, J., H. Y. Kim and L. A. Davidson (2009). "Actomyosin stiffens the vertebrate embryo during critical stages of elongation and neural tube closure." Development **136**: 677-688.

Zhou, J., H. Y. Kim and L. a. Davidson (2009). "Actomyosin stiffens the vertebrate embryo during crucial stages of elongation and neural tube closure." Development (Cambridge, England) **136**: 677-688.

Zhou, J., H. Y. Kim, J. H.-C. Wang and L. A. Davidson (2010). "Macroscopic stiffening of embryonic tissues via microtubules, Rho-GEF, and assembly of contractile bundles of actomyosin." Development **137**(16): 2785-2794.

Zhou, J., H. Y. Kim, J. H. Wang and L. A. Davidson (2010). "Macroscopic stiffening of embryonic tissues via microtubules, RhoGEF and the assembly of contractile bundles of actomyosin." Development **137**(16): 2785-2794.

Zhou, J., S. Pal, S. Maiti and L. A. Davidson (2015). "Force production and mechanical adaptation during convergent extension." Development **142**: 692-701.

Zihni, C., C. Mills, K. Matter and M. S. Balda (2016). "Tight junctions: from simple barriers to multifunctional molecular gates." Nat Rev Mol Cell Biol **17**(9): 564-580.

Zovein, A. C., A. Luque, K. A. Turlo, J. J. Hofmann, K. M. Yee, M. S. Becker, R. Fassler, I. Mellman, T. F. Lane and M. L. Iruela-Arispe (2010). "Beta1 integrin establishes endothelial cell polarity and arteriolar lumen formation via a Par3-dependent mechanism." Dev Cell **18**(1): 39-51.